PERSONALNI
računari
83
JUL/AVGUST 1992.
CENA 6000 D



OPERATIVNI SISTEMI

OS/2 2.0
KOMPLETNA PRIČA

SOFTVER

MS EXCEL 4.0 FOR WINDOWS

LOTUS 1-2-3 FOR WINDOWS

ADOBE ILUSTRATOR 4.0

NOVO KOLO NAGRADNE IGRE
COMTRAD
486 • 386 • 386SX
I JOŠ 30 NAGRADA

SPECIJALAN DODATAK NA 32 STRANE
WINDOWS 3.1
KOMPLETNO UPUTSTVO ZA UPOTREBU

VI NAMA ZNAK
– MI VAMA ŠTAMPAČ
SKRIVENI ZNAK
IZ OVOG BROJA
DONOSI
MATRIČNI ŠTAMPAČ
CITIZEN P124 D
MERKOM – KRUŠEVAC

YU ISSN 0352-7271

PERSONALNI
računari 83

časopis za informatiku i računarstvo
YU ISSN 0352-7221
Izdaje BIGZ
jul/avgust 1992 cena 6000 D
DEM 10, ATS 70,00 FRF 30,00, 500 SLT

Pažljivo pregledajte ovaj broj.
Možda je baš u njemu skriveni znak koji vam donosi matrični štampač CITIZEN 124 D sa 24 iglice - poklon firme Merkom iz Kruševca.

SADRŽAJ



Naslovna strana: Studio Tešić & Nenad Petrović. Maneken: Silvija Lazarević.

SPECIJALAN DODATAK NA 32 STRANE
WINDOWS 3.1
KOMPLETNO UPUTSTVO ZA UPOTREBU

septembarski broj časopisa računari izlazi iz štampe krajem avgusta 1992.

Sve o velikoj nagradnoj igri „Comtrad" & „Računari" na stranama 16, 17 i 19

Izdaje i štampa
Beogradski izdavačko-grafički zavod
11000 Beograd
Bulevar Vojvode Mišića 17
Generalni direktor
Ilija Rapajć
Direktor Novinskog sektora
Zoran Nikodijević
Glavni i odgovorni urednik
Jovan Regasek
Zamenik gl. i odg. urednika
Esad Jakupović
Tehnički urednik / sekretar redakcije
Natalija Desančić
Urednik za hardver
Dejan V. Veselinović
Stručna redakcija
Vesna Ćosić (aktuelnosti), Nenad Batočanin (baze podataka), Dejan Ristanović (softver), Jovan Skuljan (programiranje), Vladimir Stamenović (radne tabele), Dejan Veselinović (hardver), Dragutin Vuković (mreže), Zoran Životić (softver)
Adresa redakcije
11000 Beograd
Bulevar vojvode Mišića 17/III
Telefoni
653 748 (redakcija)
650-528 (prodaja)
Agencija EIGZ-a
(011) 651-793, 653-565
Teleks
11855 BIGZ YU
Telefaks
(011) 648-140, 647-955
Pretplata za zemlju
Za šest meseci 15000 (6 brojeva)
Za jednu godinu 27500 (11 brojeva)
(na žiro-račun D.P. BIGZ 60802-603-23264)
Pretplata za inostranstvo
Za jednu godinu 51 USD, 80 DEM, 67 CHS, 27 GBP, 271 FRF
(na devizni račun D.P. BIGZ 60811-620-16101-620701-999 03377)
Zbog usporenog prenošenja uplata preko banke, molimo pretplatnike da nam posle svake nove uplate odmah pošalju foto kopiju uplatnice.
Rukopisi se ne vraćaju.

SEZAM
sistem za modemske komunikacije
(011) 648-899 (deset linija)
Upravnik sistema
Jovan Regasek
Administratori na sistemu
Zoran Životić i Dejan Ristanović
Moderatori
Dejan Balinda (CIVILIZACIJA), Jovan Bulajić (PC USER), Aleksandar Damnjanović (PC OS), Slobodan Kalezić (ORKA), Zoran Kehler (MOST), Vladimir Kostić (PC SOFT), Vladimir Krstonošić (VICEVI), Esad Jakupovic (RAČUNARI), Dalibor Lanik (ATARI ST), Pavle Pekovič (PC PROG), Dejan Ristanović (NOVOSTI I KOMUNIKACIJE), Bojan Tepavčević (FORUM), Dejan Veselinović (PC HARD) i Zoran Životić (SEZAM)
Sekretarica sistema
Natalija Desančić
Softver
Zoran Životić (glavna aplikacija) i Dejan Ristanović (eksterni programi)
Hardver
Server: COMPAQ DESKPRO 386/25, 3 MB RAM-a, tvrdi disk 300 MB (COMPAQ Minhen). Mreža: dve stanice COMPAQ 286N/12, 1 MB RAM-a (COMPAQ Minhen), tri stanice „Ultra" AT 12 MHz. D LINK DE 100 LanSmart ("Ultra" Skopje). Modemi ViVa (Thelema Communications, Beograd)

Na osnovu Mišljenja Sekretarijata za informacije SR Srbije broj 413-01-48 od 4. 2. 1991. na ovaj časopis plaća se osnovni porez na promet po povlašćenoj tarifi od 3 odsto.

Portabli

OSEĆAJ SAVRŠENSTVA

T4400SXC je oznaka najnovijeg „Toshiba" portabl računara baziranog na Intel 486SX/25 procesoru. Računar je opremljen kolor-displejem na bazi aktivne matrice (TFT) i po specifikacijama je veoma sličan modelu T6400SXC lansiranom dva meseca ranije, ali uz jednu bitnu razliku – radi i na baterije. Specijalni slimline TFT displej koji troši 40% manje struje u odnosu na ranije modele omogućava izvlačenje tri puna radna sata iz baterija na računaru ukupne težine od samo 3.5 kg.

T4400SXC je opremljen sa 4 MB RAM-a i hard diskom od 80 MB, a osim utičnice za interni modem nema nikakvih ekspanzionih slotova. Računar se može postaviti i u „Toshiba Deck Station IV" postolje čime se pretvara u



stoni računar, a korisnici kojima snaga i brzina 486SX/25 procesora nisu dovoljni, mogu da ga zamene DX varijantom koja radi na istom taktu. Sve ovo zvuči vrlo primamljivo, što se baš ne može reći i za cenu od 5550 GBP.

Kontakt telefon: (0932) 841600.

Kartice

HARDVERSKI ANTI-VIRUS

Proizvođači anti-virusa su u poslednje vreme veoma aktivni – posebno zanimljivu novost donosi holandska firma „Novix International" (u Vel. Britaniji je zastupa firma „SCI-Clone"). Reč je o ekspanzionoj PC kartici nazvanoj „Thunderbyte" koja nadzire računar u potrazi za nedozvoljenim, najčešće virusnim, akcijama kao što su introdukcija, replikacija i aktiviranje virusa. Ove aktivnosti obično obuhvataju deo virusnog koda koji ostaje rezidentan u memoriji posle izvršavanja programa, formatiranja diska, modifikacije but-sektora ili menjanja podataka smeštenih u CMOS RAM.

Proizvođač tvrdi da je kartica potpuno transparentna za većinu aplikacija, a da pri tome vrši inteligentni nadzor serije akcija umesto pojedinačnih akcija. Poseban mod za učenje je projektovan tako da olakša „podučavanje" kartice o tome koje su sumnjive akcije normalne. Pored normalne kontrole, kartica čak proverava sistem pri samom putovanju, kako bi već u samom začetku presrela svaki potencijalni virus. Osim toga, moguće izvršiti „patch"-ovanje hard diska preko ST 506 kontrolera.

Poslednja verzija „Thunderbyte"-a, koja košta 90 GBP, je takođe i Novell autorizovana i raspoznaje 22 tipa sumnjivih akcija za razliku od ranijih 19, a podržava i SCSI, IDE i ESDI kontrolere. Proizvođač tvrdi da je osigurana zaštita ne samo protiv svih postojećih, već i protiv svih budućih virusa!

Kontakt telefon: (0284) 723289

Sporovi

AMD DOBIO SPOR SA INTELOM

Poslednja parnica u pet godina dugom sudskom ratu koji vode „Intel" i AMD („American Micro Devices") se završila tako što je AMD dobio pravo da proizvodi 386 procesore u neograničenim količinama i 15 miliona USD odštete. Slučaj je 1987. godine pokrenuo AMD u nameri da dobije pravo na proizvodnju 386 procesora. Ovome je prethodio sporazum o tehničkoj saradnji iz 1982. kojim je „Intel" prihvatio da AMD-u da prava na neke elemente svoje tehnologije, a AMD je zauzvrat prihvatio obavezu da razvije određene proizvode. Sporazum se zasnivao na sistemu poena: kada je AMD nakupio dovoljno poena, „Intel" je prestao da mu da prava na i386 čip.

"Intel", međutim, tvrdi da AMD nikada nije proizveo specificirane čipove i da zato nije dobio ni prava na i386. AMD je, sa svoje strane, posle višemesečnog čekanja krenuo samostalno i napravio sopstveni 386. „Intel", naravno, nije oduševljen presudom budući da AMD sada ima pravo na neograničenu proizvodnju 386 kompatibilnih čipova, uključujući i 40-MHz verziju, što pruža slične mogućnosti i mnogim drugim manjim firmama koje sarađuju sa AMD-om.

Procesori

KOPROCESOR NA NORVEŠKI NAČIN

Koprocesori se obično koriste za grafiku, matematiku i obradu digitalnih signala, međutim, norveški istraživači su projektovali tekstualni koprocesor. MS 160, kako glasi oznaka novog tzv. „data filter" čipa, je u stanju da pretražuje podatke ili uzorak tipa reči i rečenica brzinom od 160 MB u sekundi. Ono što je najlepše od svega je činjenica da nije potrebno podatke prethodno indeksirati. Potencijalne aplikacije ionako već uključuju pretraživanje podataka. Za rad mu je potrebno najmanje 64 MB-a, ali najbolji rezultati se postižu sa najmanje 128 MB. Isporuka će najverovatnije početi tokom trećeg kvartala ove godine, a po svemu sudeći, čip će biti ugrađivan u disk-kontrolere ili će biti kombinovan sa i860 procesorom. Cena za MS 160 ploču bez dodatne memorije će se kretati u intervalu 450-550 GBP.

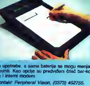
Kontakt telefon: Microway Norway, – 010-476 892 020

Ulazni uređaji

SCRIPTWRITER – OLOVKA UMESTO TASTATURE

Da li biste kupili kompjuter baziran na Z80 procesoru, opremljen sa 256 K RAM-a za 1595 GBP? Normalno da ne biste, ali „ScriptWriter II" firme „Peripheral Vision" nije „Spectrum" – reč je o *high-tech* ploči za unos podataka. Za razliku od standardnih pen-računara, „ScriptWriter" koristi običnu hemijsku olovku i običan papir, a zasniva se na ploči osetljivoj na pritisak. Sistem je u stanju da prepozna rukopis (doduše samo štampana slova) zahvaljujući posebnom softveru, a rezultati se prikazuju na malom LCD ekranu iznad papira. Ako dođe do greške u prepoznavanju ili ako korisnik pravi određene sistematske greške, pripadajući softver nudi mogućnost „učenja" i prilagođavanja rezultata. Ovakav pristup ne samo da pruža maksimalno humanizovan korisnički interfejs za unos podataka, već i automatski obezbeđuje papirni zapis svakog unosa, bez potrebe za posebnim štampanjem.

"ScriptWriter" je opremljen setom PC-baziranih programa za kreiranje različitih formulara i prenos podataka na stone računare. Radni vek baterija je deklarisan na 12 sati kontinual-



ne upotrebe, a same baterije se mogu menjati i puniti. Kao opcije su predviđeni čitač bar-koda i interni modem.

Kontakt: Peripheral Vision, (0373) 452755.

Štampači

INK-JET, A2, POSTSCRIPT – SVE U JEDNOM

Ploteri, štampači i PostScript – sve su to tradicionalno skupe stvari, naročito kada je reč o većim formatima. „ProTracer" firme „Pacific Data Products" (PDP) objedinjuje sve navedeno osim plotera – reč je o ink-jet PostScript štampaču formata do A2. Za kompaniju koja je postala poznata po proizvodnji dodataka za HP štampače, ovo je krupan ali i logičan korak napred. Novi štampač je dobrim delom baziran na tehnologiji razvijenoj za PDP-ov PostScript kompatibilni akcelerator i koristi isti i960 RISC procesor. Ink-jet mehanizam je „Canon"-ov sa maksimalnom rezolucijom od 360 dpi.

ProTracer je u principu namenjen CAD tržištu i može se koristiti direktno iz *AutoCAD*-a preko ADI (*AutoDesk Interface*) drajvera. Pored toga, tu su i Epson LQ1050 i IBM Proprinter XL24E emulacije kao standardne i HPGL (269 GBP) i PostScript (339 GBP) kao opcije. HPGL zahteva proširenje na 2 MB (199 GBP) a PostScript punih 8 MB (609 GBP) da bi mogao da se iskoristi puni A2 format. Štampač u osnovnoj varijanti košta 1049 GBP, a sa punom PostScript opremom cena iznosi nešto manje od 2000 GBP. Kao opcija je još predviđen i AppleTalk interfejs po ceni od 139 GBP, ali ovakva nadogradnja sasvim logično zahteva i PostScript. Za standardno štampanje na formatu papira do A3, korisnicima stoji na raspolaganju i opcioni uređaj za automatsko ubacivanje papira. Kaseta sa mastilom (19 GBP) je deklarisana na 150 strana A2 formata, mada ovo dosta zavisi od gustine štampe.

Kontakt: Pacific Data Products, (0442) 231414.

Softver

NOVI OCR PROGRAMI

Na tržištu su se gotovo istovremeno pojavila dva nova OCR programa, pri čemu se za svaki od njih tvrdi da donosi jedinstvenu novu tehnologiju koja poboljšava tačnost čitanja.

TypeReader firme „Formscan" raspolaže novom MLFA (*Machine-Learned Fragmentation Analysis*) tehnikom treće generacije za či-

tanje tekstova slabog kvaliteta (kao što su tekstovi u dnevnim novinama), a američka firma „Caere" se pojavila sa novom verzijom svog *Omnipage* paketa koji je opremljen novom tehnologijom za poboljšanje tačnosti čitanja pod nazivom *Anypage*.

Prema tvrdnjama predstavnika firme „Caere", *Anypage* tehnologija je, zajedno sa redizajniranom OCR „mašinom", usmerena prvenstveno na poboljšanje tačnosti OCR procesa, dok je brzina za sada u drugom planu. Ove tehnologija je nastala na bazi prethodne *Anyfont* tehnologija koja je bazirana na seriji „Eksperata" unutar softvera od kojih je svaki zadužen za prepoznavanje određenog tipa slova. *Anypage* tehnologija je opremljena mehanizmom za pročišćavanje slike pre nego što *Anyfont* mašina počne sa čitanjem. Tehnologija je prilagođena mogućnostima *gray scale* skenera visoke klase za razlučivanje nijansi tako što raspoznaje sive nijanse pozadine od nijansi samog teksta. Ova tehnologija bi trebalo prema rečima proizvođača da eliminiše čuvena noćne more koje nastaju pri prepoznavanju crnog teksta na sivoj pozadini, žućkastih dokumenata, pa čak i dokumenata sa flekama od kafe. *Omnipage Direct*, ekonomska varijanta *Professional* verzije, bi trebalo da prepoznaje tekst veličine od 6 do 72 tačke brzinom do 2.000 reči u minutu, zavisno od brzine samog računara. Pri instalaciji program pronalazi sve relevantne programe na računaru i instalira se kao **Scan** opcije **Main File** menija. Program radi samo sa *Flatbed* skenerima, budući da se ručni skeneri smatraju nedovoljno preciznim i pouzdanim za ove potrebe.

TypeReader, sa druge strane, je opremljen *Adobe Type Manager*-om i takođe koristi MLFA tehnologiju koja radi tako što posmatra unutrašnje konture slova umesto spoljašnjih, što olakšava prepoznavanje karaktera sa oštećenim ili nejasnim ivicama kao i kerniranih karaktera. Program je demonstriran na jednom tipičnom tabloidu štampanom na običnom rotopapiru, što znači prilično slabog kvaliteta, a procenat pogodaka je bio začuđujuće visok. *TypeReader* radi tako što prihvata celu stranu skeniranu na flatbed skeneru, razdvoji grafiku od teksta (bit po bit) a zatim prelazi na samo prepoznavanje. WYSIWYG editor prikazuje dokument na ekranu onako kako bi se pojavio na papiru i dozvoljava korisniku da unosi izmene. Proizvođač tvrdi da bi program trebalo da radi podjednako brzo na 286 i 386 mašinama. Razlog tome je činjenica da je dizajniran za 286 mašine, što znači da ubrzanja samog procesora nemaju mnogo uticaja. Kompanija je najavila i punu 386 verziju uskoro.

OmniPage košta nešto ispod 450 GBP, dok *TypeReader* košta 795 GBP.
Kontakt: Computers Unlimited (Omnipage), (081) 200 8282
Formscan (TypeReader), (0373) 614446.

ZAKONODAVSTVO — ZAŠTITA LIČNIH PODATAKA

I TO JE DEMOKRATIJA

Nagli prodor informatike, zasnovan pre svega na izvanrednom razvoju tehnologije u proizvodnji računara i telekomunikacione opreme, doveo je i do masovne pojave ličnih podataka u raznim evidencijama, registrima, katalozima, imanicima i sličnim zbirkama podataka.

U nastojanju da se obezbede odgovarajući pravni instrumenti u pogledu zaštite prava svakog pojedinca na zakonitost, privatnost i tajnost podataka koji se na njega odnose, većina zemalja Evropa donela je svoje zakone o zaštiti ličnih podataka. Na primer, u zakonodavstvu Savezne Republike Nemačke još od 1977. godine prisutan je i ovaj zakon.

Sve se više uočava i potreba razmene ličnih podataka, u prostorima koji često prevazilaze granice jedne zemlje. Kako se u zakonodavstvima pojedinih zemalja različito regulišu pitanje zaštite ličnih podataka, a često i ne postoje propisi koji ovu oblast regulišu, Evropski savet je januara 1981. predložio Konvenciju br. 108 o zaštiti ličnosti u pogledu automatske obrade ličnih podataka, sa namerom da je sve zemlje članice ratifikuju i usklade svoje zakone i propise sa osnovnim načelima i preporukama iz ovog dokumenta.

S obzirom na stalno usavršavanje tehničkih sredstava i drugih informatičkih resursa, zaštita ličnih podataka je dinamički proces, koji se konstantno i organizovano prati i usavršava od strane mnogobrojnih stručnjaka međunarodne zajednice. Tako je septembra 1990. nastao novi dokument Komisije Evropske zajednice „Direktive Evropskog saveta vezane za zaštitu ličnosti u obradi ličnih podataka".

Na našim prostorima pitanje zaštite ličnih podataka nije do sada bilo regulisano saveznim propisima. U Zakonu o osnovama društvenog sistema informisanja i informacionom sistemu federacije (zakon koji ništa ne reguliše i ničemu ne služi) samo je nagoveštena potreba za ovom zaštitom. U Zakonu o DSI Republike Srbije, članovi 24 i 25 daju neke elementarne osnove zaštite prava ličnosti u pogledu ličnih podataka i njihove obrade. Na prostorima bivše Jugoslavije samo Republika Slovenija ima zakon o zaštiti ličnih podataka donet u 1990. godini i usklađen sa Konvencijom Evropskog saveta.

Neka od osnovnih načela zaštite ličnih podataka sastoje se u sledećem:

- Zaštita ličnih podataka se obezbeđuje postupcima i merama kojima se sprečava njihovo nezakonito prikupljanje i korišćenje. Lični podaci o građanima mogu se prikupljati i koristiti samo na osnovu zakona ili na osnovu izričite pismene saglasnosti subjekta.
- Građanin treba da ima pravo uvida u podatke koje se o njemu vode u raznim zbirkama i da zahteva ispravke za podatke koji nisu tačni.
- Pojedini lični podaci koji se odnose na rasno poreklo (kod nas možda i na nacionalnu pripadnost?), religiozne i druga verovanja, politička opredeljenja, zdravstveno stanje i seksualni život mogu se prikupljati samo uz izričitu saglasnost osobe na koju se odnose.

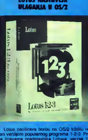
Imajući u vidu Srbiju u kontekstu značajnih političkih i ekonomskih transformacija sa deklarisanim tendencijama približavanja Evropskoj ekonomskoj zajednici, uprkos značajnog udaljavanja u praksi, jedna od značajnih zakonodavnih aktivnosti je i donošenje Zakona o zaštiti ličnih podataka, poštujući pri tom preporuke i dostignuća razvijenih zemalja Evropa, čime bi se operacionalizovao još jedan od neophodnih činioca zaštite ljudskih prava u savremenim demokratskim društvima.

Problem zaštite ličnih podataka predmet je mnogih međunarodnih stručnih skupove i sastanaka, na kojima bi trebalo intenzivirati naše prisustvo i organizovati odgovarajuća aktivnosti u zemlji, u cilju kontinuiranog i usaglašenog pristupa i daljeg razvoja mera zaštite ličnih podataka, kao jedne od bitnih pretpostavki budućih integracija.

M. Pavlov

Radne tabele

LOTUS NASTAVLJA ULAGANJA U OS/2



Lotus zaoštrava borbu na OS/2 tržištu novom verzijom popularnog programa 1-2-3. Prema izjavama predstavnika Lotusa, verzija 1.1 za OS/2 ima znatno bolje grafičke mogućnosti kao i višedimenzionalnu tabelu koja vam dozvoljava da koristite treću dimenziju u svojoj tabeli. Lotus je nastavio rad na 1-2-3 za OS/2, uprkos teškoćama, i razvio novu verziju za OS/2 verzije 2.0, a, za budućnost, obećana je i odgovarajuća verzija za sve mašine i sve operativne sisteme. Kompanija je, takođe, počela sa prodajom server verzije za 1-2-3 za *Windows*. Košta 635 britanskih funti, isto kao i DOS server verzija, dok, pak, *Windows* verzija za jednog korisnika košta 475 funti. Po novoj licencnoj šemi kompanije isti paket će se prodavati sa popustom od 10% za pet, odnosno, 20% za dvadeset korisnika. Dalje, korisnici DOS verzija mogu izvršiti nadogradnju za 99 funti po kopiji. Ista je cena i za rad u mreži, ali se mora kupiti i server verzija za 1-2-3 za *Windows*. Nakon odustajanja od razvoja programa za OS/2 2.0 Microsoft se drži po strani i dosta rezervisano posmatra šta se dešava sa novom verzijom 1-2-3 za OS/2. Microsoftov predstavnik demantuje glasine da rade na verziji *Excel*-a za OS/2, a ako bude nova verzija OS/2, ona će, u svakom slučaju, pokretati *Windows* verziju *Excel*-a.

Ulazni uređaji

TRODIMENZIONALNI MIŠ

Poznati proizvođač miševa i ulaznih uređaja uopšte „Logitech" nudi novi upravljački uređaj koji je zamišljen kao razvojno oruđe za trodimenzionalnu televizijsku grafiku. „3-D Mouse" je predviđen za primene u računarskom dizajnu i oblikovanju veštačkih računarskih prostora (*Cyberspace*). Kako tvrdi proizvođač, arhitekti i ostali inženjeri mogu pomoću novog uređaja da koriste ekran monitora kao pano. Kretanje u prostoru, odnosno po sve tri prostorne ose, se registruju putem ultrazvuka, podrazumeva se da je moguće vršiti i potpuno pravolinijske pokrete, kao i okretanje. Očitavanje položaja se obavlja 50 puta u sekundi u vidu merenja rastojanja između mikrofona i zvučnika.



„3D Mouse" je, zahvaljujući ergonomskom dizajnu, podjednako dobro prilagođen i dešnjacima i levorukim korisnicima. Očekuje se da će redovna proizvodnja početi tokom leta, a u međuvremenu se direktno od švajcarskog ogranka „Logitech International"-a za oko 1 000 USD može kupiti jedna ograničeno razvijena osnova, zajedno sa prototipom miša.

Štampači

PANASONIC: NEČUJNI ŠTAMPAČI

Nova tehnologija proizvodnje glava matričnih štampača svela je inače po zlu čuveni nivo buke koju oni proizvode na nivo ink-jet ili laserskih štampača – tako bar tvrdi „Panasonic", vlasnik nove tehnologije i proizvođač tri nova 24-iglična modela. Sva tri modela postižu rezoluciju do 360×360 dpi, tri umesto standardna dva nivoa kvaliteta ispisa (draft, tekst i LQ), po tri fonta u draftu, šest u tekst-modu i po jedan u LQ načinu.

Ekonomski model sa oznakom KX-P2123 (cena 848 maraka) u draft modu postiže brzinu do 192 znaka u sekundi. Sledeći model KX-P2124 (cena 998 maraka), po izboru fontova sličan prethodnom modelu, postiže u istom modu brzinu ispisa do 256 znakova u sekundi. Za oba modela se opciono može nabaviti i uređaj za automatsko ubacivanje pojedinačnih listova papira i kolor-kit, a standardno su opremljeni traktorom za beskonačni papir i uređajem za pozicioniranje i odsecanje papira. Najjaču ponudu u 24-igličnoj gami predstavlja model KX-P2624, koji pored svega navedenog radi sa papirom do formata A3, opremljen je LCD-displejem i košta 1600 maraka.

U novu seriju „tihih" štampača spada i jedan 9-iglični model KX-P2180 (cena 648 maraka), kod koga je nizak nivo buke postignut posebnom izolacijom kućišta; ovaj model u draft modu postiže brzinu do 192 znaka u sekundi.

Procesori

486DX2 ILI 50 MHz U SERIJSKOJ PROIZVODNJI

„Intel" je predstavio novi procesor sa oznakom i486DX2-50 kao suštinsku novost u familiji i486 procesora. DX2-procesori se zasnivaju na novoj „Intel"-ovoj tehnologiji koja omogućava dupliranje internog radnog takta procesora. Spolja gledano, odnosno sa stanovišta ostatka sistema, takt ostaje nepromenjen i iznosi klasičnih 25 MHz. Rezultati „Intel"-ovih testova, u kojima je novi procesor upoređen sa standardnim 486-procesorom, pokazuju povećanje brzine od oko 70 do 100%. Procenjuje se da bi takvo ubrzanje mogli da donesu samo puni 50 MHz sistemi i pitanje je da li će u praksi ova tehnologija zaista doneti obećani nivo ubrzanja. Iako ostatak sistema i dalje radi na 25 MHz, DX2 procesori ipak postavljaju velike zahteve, pre svega pred BIOS i memoriju. Procenjuje se da će od nove tehnologije najviše profitirati proizvođači modularnih PC računara.

„Intel" će svoju paletu 486-procesora tokom godine proširiti na tridesetak modela, što donekle predstavlja razlog za brigu da ovo mnoštvo tipova ne dovede do velike zbrke. Već danas se proizvodne politike na tržištu računara i procesora, uzimajući u obzir manje i realnije potrebe kupaca, postavlje kao strateška orijentacije marketinga. Prihvatanjem i386 kao procesora za kućno računarsko tržište, „Intel" pokušava da dostigne svoj raniji monopolski položaj, dok povećanom aktivnošću na 486 segmentu tržišta i novim modelima teži da što više potisne neugodna i sve jače konkurente na 386 segmentu tržišta. Jednu od krupnih smetnji predstavlja velika raznolikost modela nastale nadgradnjom istih tipova procesora raznih proizvođača. Veštom marketinškom strategijom i pažljivim određivanjem proizvodne politike bi se mogao staviti pod kontrolu veći deo tržišta.



Takva politika proizvodnje, međutim, ne može biti u interesu kupaca, jer rat cena razvijenih i sve bržih modela koji se puštaju na tržište u kratkom roku, kao izraz monopola jednog proizvođača, po pravilu vodi ka lošijim proizvodima. Jedna predimenzionirana raznolikost modela povlači, takođe, još drugih problema, kao što je otežana mogućnost efikasnog pregleda karakteristika svih verzija i486 procesora. Trgovci će se konačno odlučiti da prodaju samo mašine koje im omogućavaju najveću maržu, dok kupac pri tome gubi mogućnost objektivnog izbora modela koji sve više prepušta trgovcima, a time postaje neizvesna i mogućnost kontrole. Iz ovoga se nameće zaključak da sada prodajna strategija postaje važnija od čisto tehničkog aspekta.



SKOK KROZ HIPERSVEMIR

Malo po malo, procesor 80486 se uspinje po skali performansi. Sa početnih 25 prvo se popeo na 33 MHz, a sada se već prodaje i na 50 MHz. Koliko je to brzo, kome to treba, koliko košta – to su pitanja na koje ovde dajemo odgovore.

Dejan V. Veselinović

Nikoga ne treba da čudi proces daljeg razvoja sada već dobro poznatog procesora i80486DX. Prvo su od njega napravili ogoljenu verziju i nazvali ga „SX" (za „single Sigma"), sada spremaju verziju sa dvostrukim radnim taktom (jednim za sam procesor a drugim za komunikaciju sa periferijama), drugi pripremaju klon procesora (C&T „NextGen", IIT „Cyrix", AMD), a cene gotovih proizvoda padaju i padaju, na žalost ne onoliko brzo koliko naša kupovna moć. Konačno, i sam „Intel" se sprema da prikaže svoj najnoviji procesor pod radnom oznakom „P5", u javnosti poznat i kao 80586.

U međuvremenu treba nešto i prodavati. Pošto je procesor 80486 napravljen u submikronskoj tehnologiji, što znači da je razdaljina između veza manja od jednog mikrona, i pošto je od samog početka bio projektovan tako da teorijski može da radi na taktovima do 150 MHz, „Intel" je sasvim logično i očekivano na tržište izbacio verziju ovog procesora na 50 MHz, najveći radni takt koji su ikada komercijalizovali. Naravno, problema je bilo, prve serije su bile povučene usled proizvodnih grešaka koje su dovodile do nestabilnosti, pa su firme kao „Compaq" i „Dell" morale da vraćaju prodate primerke. Sada su problemi rešeni i prodaja teče normalno.

50 MHz je zapravo 51% brže od dosadašnjeg plafona na 33 MHz. Za tu brzinu treba da izdvojite i dosta para, u ovom trenutku između 500 i 700 DEM za ovu razliku. Čisto ekonomski gledano, to je oko 20% više na cenu cele mašine u nekoj baznoj konfiguraciji, što znači da 51% veće procesorske performanse plaćate 20 ili manje posto cene, odnos koji sam po sebi opravdava razliku – ako vam treba ovakva računarska snaga.

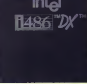
Daleko interesantnije pitanje u sadašnjem trenutku jeste da li ovakve platforme zaista i ostvaruju očekivanu razliku u performansama kada se zna da sabirnica i dalje radi na ubogih 8 MHz sa jednim stanjem čekanja, uz sve ostale probleme koji iz toga proizlaze. Ipak, jedno je procesor, a drugo je još uvek nužna mehanika, koja je spora i za daleko sporije procesore.

Pogledaćemo dve mašine iz dva izvora koje sada odmah možete kupiti ako vam trebaju. Veoma su različite po svemu sem po procesoru, što je veoma dobro, jer prikazuju mogućnosti nove platforme u dva segmenta kojima je prevashodno i namenjena.

IMTEL 486-50 MINI

Odmah da se ogradimo – ime modelu smo sami dali. „Imtel" je inače skraćenica od „Instituta za mikrotalasnu elektroniku", a nalaze se na Novom Beogradu. U kontekstu računara, ova firma spada u dobro poznate izvore ličnih računara, a ima i jednu specifičnost – koliko je nama poznato, ni jedan drugi prodavac računara na našem tržištu nema takvu strukturu zaposlenih, sa nekih preko 70% elektroinženjera, od razvoja do prodaje.

Ovaj model smo nazvali „mini" samo zahvaljujući kućištu koje je zaista mini formata, mada ne onako malo kao „booksize" model firme O.T.C. Po širini može da primi dve jedinice



Snimio: Studio Tesic

Na vrtoglavom taktu: Mikroprocesor i486/50 MHz

od 3.5 inča, a kartice za proširenje se mogu staviti u horizontalan položaj (paralelno sa matičnom pločom). Mesta ima za ukupno 3 16-bitne i jednu 8-bitnu karticu; u našem uzorku, dve 16-bitne utičnice su bile popunjene video karticom i kombinovanom I/O karticom i AT Bus adapterom.

Ovakva koncepcija je, naravno, uslovila i niz pratećih rešenja: recimo, izvor napajanja ima potpuno nestandardan oblik, duguljast je i uzan, a izduvni ventilator je mali i nalazi se iznad, a ne u samom izvoru napajanja, čime se snaga izvora ograničava na 60 VA. Ovo nije baš mnogo, ali bi trebalo da bude dovoljno za veliku većinu mogućih konfiguracija u ovakvim kućištima. Najzad, najveći potrošači su tvrdi diskovi, a po samoj arhitekturi kućišta ne možete ih staviti više od jednog ako mislite da zadržite flopi disk jedinicu.

Ovo u vezi flopi disk jedinica je, inače, i naša jedina stvarna zamerka ovom kućištu. Tako je projektovano da je nemoguće u njega ugraditi standardnu AT disk jedinicu od 5.25 inča (ili bilo koju drugu jedinicu ovog formata). Možemo mi da pričamo kako je taj format disketa prevaziđen, što tehnički gledano zaista i jeste, ali ovo smatramo manom iz razloga praktičnosti – nema AT mašine bez te disketne jedinice, ali zato nema baš previše modela koji imaju disketnu jedinicu od 3.5 inča.

Naravno, kućište, a bogami i model su namenjeni jednom dosta specifičnom profilu korisnika, pa u tom kontekstu ova mana donekle gubi na značaju. Na kraju krajeva, ako nekome baš treba, uvek može da se opredeli za neko standardnije kućište. Nama se inače ovo malo kućište zaista veoma dopalo – bogom je dano za radne stanice i za one kojima je prostor na radnom stolu veliki problem. Dizajn je čak veoma prijatan, a mehanička konstrukcija veoma dobra. Što se samog kućišta tiče, ako vam trebaju male dimenzije, a ne marite za disketne jedinice od 5.25 inča, kupite ga – odlično je.

Tvrdi disk je poznati model firme „Western Digital" „Piranha 4200", flopi disketnu jedinicu je potpisala firma „Epson", a video kartica koristi „Tseng" ET4000 video čip sa 1 MB memorije (80 nS) i „Sierra" RAMDAC čipom, koji obezbeđuje potrebnih 33.000 boja u rezolucijama od 640×480 i 800×600.

Sve to se prikazuje na jednom „Panasonic" C1381 monitoru koji nam se veoma dopao. Ima stabilnu sliku, dobro koristi raspoloživu površinu prikaza, odnosno ima male bordure crnog oko slike. Naravno, ne može se porediti sa visokom klasom, a „Imtel" nudi i širok izbor većih i boljih monitora, sve do onih sa dijagonalom od 21 inča (53.3 cm). Probali smo i ovaj veliki – šta da vam kažemo, ostanete bez daha. Sa 33.000 boja u rezoluciji od 800×600 TV prijemnik deluje naprosto nekako – anemično.

Matična ploča je veoma interesantna po koncepciji. Spada u red „dibidus mini" ploča, što znači da je manja čak i od „Baby AT" formata. Uprkos tome, na njoj se nalazi 8 SIM modula sa podnožjima za SIP module (!) između njih. Time se na ploči može montirati ukupno 64 MB memorije, što u ovakvim slučajevima nije neverovatna količina. Tu je i šest 16-bitnih utičnica, ali nigde nema podnožja za numerički koprocesor (recimo od firme „Weitek") i nigde nema spoljne keš memorije.

Koncepcija ploče se svodi na „malo je lepo", pa su se projektanti opredelili da se oslone isključivo na onih 8 KB keš memorije ugrađenih u sam procesor. Pored uštede na prostoru, ovo predstavlja i uštedu u ceni, jer statička memorija nije jeftina, posebno na nivou brzina potrebne za ovaj radni takt. Na ovom nivou, sama keš memorija bi trebalo da radi na 15 ns, a tag RAM na 10 ns, trenutno retkim brzinama sa velikim cenama. Sistem, naravno, radi i bez drugostepene keš memorije, ali se prirodno postavlja pitanje kako, odnosno koliko brzo, posebno u poređenju sa sporijim mašinama na 33 MHz i, naravno, konkurencijom koja ima drugostepenu keš memoriju. O tome malo više kasnije.

GAMA 486-50 SERVER

Opet je naziv modela naš. Ne znamo kako bismo ga drugačije nazvali – radi se o impresivnom „tower" kućištu, obojenom u mat crnu boju. Prvo ime koje nam je palo na pamet je „Darth Wader", pa bi onda „Imtelov" model mogao biti „Luke Skywalker", jer ovo su ionako „Zvezdani ratovi" bar po performansama.

Koncepcija „Gamine" mašine je potpuno suprotna „Imtelovoj". Kućište je veliko, izvor napajanja gotovo četiri puta jači, kapacitet diska je 2.5 veći, a sa 512 MB malo kome će trebati još, sem onih kojima je potreban server. Disk je naravno „Fujitsu" (o njemu smo već pisali ranije), video karta koristi čip WD 90C30 sa 1 MB video memorije, a njena granica po broju boja je 256. Najzad, kućište je imalo ugrađenu samo jednu disk jedinicu od 5.25 inča, što je opet nedovoljno u kontekstu namena. Objašnjeno nam je da očekuju male disk jedinice u crnoj boji i nisu hteli da ugrade jednu belu jedinicu u ovako lepu crnu kutiju.

Monitor odabran za ovu priliku koji „Gama"



Za CAD/CAM profesionalce: Imtel 486/50

Snimio Studio Tešić

nudi kao deo standardne ponude za ovaj model je novi „Philips" super VGA monitor iz serije „Brilliance". Do sada je to bilo slatko tržišno ime a sada i stvarnost. Boje treba da vidite – toliko su jasne i čiste da bez ikakve griže savesti ili rezerve možemo da konstatujemo da je to najbolji monitor koji smo ikada do sada videli u tom formatu (klasa 14-15 inča). A taman smo zaključili da nas ništa ne može iznenaditi...

Koncepcija matične ploče je sasvim drugačija od one koju koristi „Imtel". Radi se o „Baby AT" formatu ploče koja u svemu nalikuje slanjim modelima i koja poseduje spoljnu ili drugostepenu keš memoriju od 64 KB (proširivo do 256 KB). Ploča prima do 64 MB memorije u osam SIM modula i dva posebna SIM modula sa dvostrukom širinom. Ima ugrađenu podršku za dvostepeni radni takt pa ćete moći da zamenite procesor sa onim koji će interno raditi na celih 100 MHz (ako ga ikada bude).

Slika 1



Mašinski testovi

Slika 2



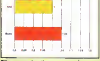
Programski testovi

Slika 4



Ocene u rasponu od 1 do 10

Slika 3



Dijagram vrednosti

NA BRZINSKOM TESTU

Kao što se iz priloženih tabela vidi, dobili smo veoma interesantne rezultate. Uzgred budi rečeno, rezultati su dati prema najnovijoj verziji naših „Unitestova", sada već sa oznakom 4. Primetićete nekoliko razlika. Prvo, nema prikaza brzina prenosa, pošto ona zavisi od specifične veze računar-disk, i pošto su ti rezultati faktički dati u okviru performansi tvrdog diska, smatrali smo ih potpuno suvišnim. Drugo, primatićete jedan dodati test u okviru paketa programa, onaj za *Windows*, njega smo uveli zbog sve većeg prelaska na rad pod ovim programom. I treće, tu su dva (za svaku mašinu po jedan) nova dijagrama, dobijeni su na osnovu naše procene niza faktora i prikazuju koliko neki proizvod u kvalitativnom smislu odgovara svojoj ceni.

Pomak u brzini koji ove mašine nude u odnosu na našu referencu, SX na 25 MHz bez keša, faktički je veći nego što je svojevremeno AT nudio u odnosu na PC/XT. Tada je razlika bila oko 2,62:1 (AT je tada radio na 6 MHz), a sada je najmanja razlika dala indeks od 3,32:1 u mašinskim testovima (za 486 na 33 MHz). „Imtel" je na mašinskim testovima zaradio indeks od 2,55, a „Gama" 3,69. Iako je „Gama" konzistentno bila brža od „Imtela", osnovni razlog ovako velikoj razlici se nalazi u video odeljku: kartica koju „Gama" koristi (WD 90C30) poprilično je brža od „Tseng" ET4000. Zauzvrat, „Imtel" nudi 33.000 boja prema 256 kod „Game".

Druga razlika je nastala na brzini rada sabirnice. Pošto smo mašine dobili kao gotove proizvode, u principu nismo ništa dirali. Obe mašine su bile podešene na maksimum mogućeg. U slučaju „Imtela" ne znamo koliko je to tačno, ali je u SETUP-u navedeno kao „high speed", kod „Game" sabirnica je radila na 16,5 MHz.

U prvom slučaju, to navodno brzo i nije mnogo brzo, na merenjima smo dobili propusnost od 2,7 MB/s, što odgovara brzini od oko 6,3 MHz, to uopšte nije brzo, to je standard, pa utoliko smešnije izgleda upozorenje u priručniku da ima kartica koje neće hteti da rade na ovoj brzini? U drugom slučaju, neki ljubitelj naučne fantastike je, verovatno pod uticajem one čuvene rečenice iz „Zvezdanih staza" „Warp speed Mr. Spock!" („Brzina za hiperkosmos gospodine Spok!") krenuo sa gasom do daske, što se takođe pokazalo lošim, jer nismo mogli da raspakujemo polovinu datoteka sa disketa (uz stalnu poruku „Packed file corrupt!"). Zato smo brzinu sabirnice na „Gaminoj" mašini morali da smanjimo na 12,5 MHz, što bismo i njima savetovali. To je opet 56% brže od standarda, ali sve je normalno radilo.

Brzina sabirnice se najviše oseća na brzini videa, što brža sabirnica, brži video odziv. Mi na našem ličnom računaru sabirnicu držimo na 13,3 MHz. Na tom taktu protok video podataka se kreće od 6,2 MB/s za pisanje do 3,1 MB/s za čitanje, dok je protok na sabirnici 5,2 MB/s za čitanje i 4,3 MB/s za pisanje. Ove vrednosti su bar 60% brže od najbržeg na „Imtelovoj" ploči, koja jeste da je u okviru standarda, ali je ipak malo spora za svoju klasu.

Iz svih tih razloga „Imtel" je na video testovima zaradio indeks od 2,55, prema 5,94 za „Gamu", ili 6,20 za naš 486 na 33 MHz. U ostalim testovima, „Gama" je povela za prosečno 12% po testu.

A onda smo stigli do programskih testova. E, tu je tek postalo veselo. Odmah da kažemo da u testu sa bazama podataka zaprevo i nije bilo neke razlike, „Imtel" je zaradio indeks od

1,72, a „Gama" indeks od 1,73 Time se razlike u kvalitetu diska zapravo pokazuju kao potpuno zanemarive

Na testu sa obradom teksta „Imtel" je dobio indeks od 3 44 (odličan), a „Gama" 3 70 ili 7 6% brže U testovima sa poslovnom grafikom, razlika je već bila veća (30% razlike u korist „Game" 3,81 za „Gamu" prema 2 93 za „Imtel") U slučaju tabelarnih proračuna, „Gama" je povela za 13% ispred „Imtela" (4,45 prema 3 94)

Najzad dolazimo na ona dva odeljka kod kojih video igra presudnu ulogu podsećamo da je „Imtel" ovaj računar namenio pre svega CAD/CAM aplikacijama. U slučaju CAD testove, „Gama" je povela za skromnijih 6 8%, što jeste prednost, ali je dosta mala Međutim, u slučaju testova za *Windows*, pre svega zbog brže sabirnica i brža video karta „Gama" je povela za celih 48 6%, što je i te kako osetno bez ikakve štoperice Sasvim očekivano, najveće razlike su nastale na testovima koji zahtevaju brzo prebacivanje podataka sa i na karticu (brža kartica, brža sabirnica), kao, recimo pomar teksta na ekranu („Imtel" 901 znak/sekundi „Gama" 1 692 znaka/sekundi)

Sva u svemu, u kontekstu apsolutnih performansi „Game" je povukla za nekih 15 8% ispred „Imtela" Ova razlika potiče od veoma brze video karte pre svega a potom iz očigledno veće brzina sabirnice Najzad „Gama" ipak nudi drugostepenu keš memoriju a „Imtel" ne pa čak i na čisto procesorskim testovima „Gama" prednjači za nekih 8 3%, razika u brzini prilikom pristupa memoriji od 21 7% u korist „Game" samo potvrđuje ovu tezu

U poređanju sa apsolutnom referencom, obe mašine nude i te kako osetan porast performansi ako prelazite na ovu klasu sa recimo klasične AT mašine (80286 na 16 MHz), očekujte ubrzanje od oko 5 1 Subjektivan osećaj je isti kao kada biste u „Jugo 45" ubacili motor BMW-a sa ubrizgavanjem


Snimio Studio Tešić

Server se velike mreže: Gama 486/50

Lična karta

MODEL	Imtel	Gama
Matična ploča		
– Procesor	80486	80486
– Koprocesor	Podržan	Podržan
– Radni takt	50 MHz	50 MHz
– Keš memorija	– standardne	--
64 KB		
– proširivo do	--	256 KB
– Sistemska memorija	– standardne	8 MB
8 MB		
– proširivo do	64 MB	64 MB
Tvrdi disk		
– Proizvođač	WD	Fujitsu
– Model	4200	M2624T
– Kapacitet	210 MB	512 MB
– Radi pod DOS verzijom	5 0	3.31
Video podsistem		
– Monitor proizvođač	Panasonic	Philips
– Monitor model	C1381	Brilliance
– Dijagonala cev	14"	14"
– Maks. rezolucija	1024×768i	1024×768i
– EM zračenje	Malo	Malo
– Video karta proizvođač	?	?
– Video karta model	?	?
– Video čip	Tseng ET4000	WD 90C30
– Memorija	1 MB	1 MB
– Maks rezolucija	1024×768	1024×768
– Maks. broj boja	33.000	256
Ostalo		
– Flopi disk jedinica, 5,25"	–	TEAC
– Flopi disk jedinica, 3,5"	Epson	Opciono
– Paralelnih linija	1	1
– Serijskih linija	2	2
– Game linija	–	
– Modem kartica	–	
– Faks/modem kartica	–	–
– Tastatura	Chicony	Cherry
Prodavac		
– Prodavac	Imtel	Gama
– Adresa	Bul.Lenjina 165b 11000 N Beograd	Milerska 3 11070 Beograd
– Telefon (011)	134-516	332-275
– Telefaks	138-928	335-902

SVAKO U SVOJOJ KLASI

Oba modela se mogu smatrati uspešnim u segmentima kojima su namenjeni Model koji nudi „Imtel" je zamišljen kao CAD/CAM stanica, tako je i opremljen tu ulogu igra sasvim dobro, što znači nešto malo sporije od modela „Game", a opet mnogo brže od svoje sporije sabraće Čak su i dimenzije kućišta tako podešene da što bolje ispuni osnovnu namenu Prema tome, smatramo da je cilj postignut, ovo je model namenjen CAD/CAM profesionalcima.

"Gama" se opet okrenula onime koji žele veoma jak server I na tom računaru je sve prilagođeno toj potrebi, praktično, nema mesta bilo kakvoj zamerci, sem optimističkog podešavanja sabirnice na preveliku brzinu, što se naravno lako da promeniti u inače odličnom MR BIOS-u Pretpostavljamo da ća „Gama" ispuniti obećanja i ovoj mašini dodati crnu flopi disk jedinicu formata 3,5 inča

Dakle ako tek treba da se opremite, obe modela predstavljaju odličan izbor u svojim klasama „Imtel" nudi pošten posao uređaj koji vredi koliko traže za njega a „Gama" je u ovom slučaju ponudila i jednu natprosečnu vrednost Neka vas ne zbuni malo prebacivanje iznad jedinice kada se malo bolje zamislite svako prebacivanje ili podbacivanje nije zanemarivo uz pretpostavku da vam niko ne poklanja računar ili baš otvoreno krade pare Ako već posedujete 486 na 33 MHz, ne verujemo da ima smisla menjati ga, pa čak i ako radi na „svega" 25 MHz. Za sve ispod toga, ima smisla razmisliti – što niže, to manje mesta razmišljanju

OCENA

Evo našeg viđenja učesnika

Ocene

	Imtel	Gama
Kućište (1-10)	9	8
Tastatura (1 10)	7	8
Monitor (1-10)	10	10
Elektronika (1 10)	8	10
Sklapanje rad (1-10)	10	10
Garancija (1-10)	10	10
Performanse		
– apsolutne (1 10	8	10
– relativne (1-10)	8	10
Prosečna ocena	8	8 5
Cene		
– apsolutna (1 10)	9	10
– relativna (1-10)	8	8
Indeks cene (prosek)	8	8
VREDNOST	1 0	1 05

U tabeli ocena pojam relativnog se odnosi na prosek klase kojoj uređaj pripada, dakle, radi se o poređanju mašina sa prosekom sličnih modela (kada je to moguće) pored već uobičajenog poređanja sa apsolutnom referencom

Apsolutna cena se određuja prema referenci a relativna cena prema proseku klase (pet beogradskih prodavaca) Vrednost dobijamo kada prosečnu ocenu podelimo sa indeksom cene vrednost od 1 označava da proizvod objektivno vredi ono što se za njega traži (fer posao korektna cena u odnosu na kvalitet) Ako kod vrednosti dobijemo brojku manju od 1 onda je cena neodgovarajuća vrednosti, odnosno plaćate više nego što dobijate i obratno – ako je vrednost veća od 1, dobijate relativno više nego što plaćate (odnosno radi se o odličnom poslu) Usled dnevnih kolebanja na tržištu proračunati odnosi su tačni na dan proračuna (20 04 1992), a svim zainteresovanim savetujemo da se pre kupovina raspitaju kod prodavaca

Obe mašine su dobile iste ocene za svoja kućišta jednostavno zato što su oba odlično napravljene zamišljena i obrađena mada pripadaju različitim klasama odnosno namenama Smatramo da je „Cherry" ipak bolja tastatura od „Chiconyja" pa otud i razlika u bodovanju uz napomenu da je ova verzija „Chicony" tastature bolja od prethodnih sa istom oznakom (KB-5191)

Svaki monitor je odličan za svoju klasu pa zato imaju isti broj bodova U kontekstu elektronika jasnu prednost dajemo ploči koju prodaje „Gama" smatramo da je pokazano da se na ovom nivou brzine ne može bez sekundarne keš memorije sem možda za CAD koji se gotovo u celini oslanja na procesor i maltene ni na šta drugo

Kvalitet sklapanja je u oba slučaja odličan se urednim montiranjem povezanim kablovima i čvrsto montiranim delovima Isto važi i za garanciju, po 24 meseca u oba slučaja. Pošteno zasluženi maksimalni poeni

O performansama je već bilo reči u apsolutnom smislu, oba modela su korak napred u odnosu na sve ostalo U relativnom smislu, „Gama" nudi veću fleksibilnost i bolja performansa

TRKA BEZ GRANICA

"Računari" su tokom poslednjih godina prikazali veliki broj najrazličitijih PC konfiguracija, od jeftinih AT klonova do najmoćnijih mašina zasnovanih na poslednjoj generaciji Intelovih mikroprocesora. Nema nikakve sumnje na kojoj se strani nalazi računar SC4050 firme „Šutlić" koji ovde testiramo – 80486 na 50 MHz, 8 M RAM-a, SCSI kontroler sa 4 M keš memorije na sebi i drugi dodaci naprosto će vas naterati da se zapitate gde su granice PC tehnologije!

Dejan Ristanović

Kada smo već pomenuli kućište da počnemo baš od njega sav opisani hardver je „upakovan" u MIDI kućište (možete ga postaviti kako na sto tako i pored njega) u kome i dalje ima začuđujuće mnogo mesta – možete da smestite tri disk jedinice od 5.25 inča standardne visine i još 3 disk jedinice od 3.5 inča svakako više nego što će vam biti potrebno U mašinu koju smo testirali bila je ugrađena disk jedinica od 5.25 inča i *Maxtor* hard disk 3.5 inča/210 megabajta, verujemo da se u komercijalnu verziju ugrađuje i flopi disk jedinica od 3.5 inča dok su preostale „fioke" rezervisane za strimer, drugi hard disk i ostale dodatke koje ćete vremenom nabavljati Kućište ukratko dobija visoku ocenu kako za funkcionalnost tako i za izgled

MEMORIJA NA KVADRAT

Što se elektronike tiče, ugrađena matična ploča je *baby* dimenzija (ploče se više na smanjuju iz godine u godinu, nego iz meseca u mesec), na njoj dominira mikroprocesor 80486 koji radi na 50 MHz i sa sistemskom memorijom komunicira posredstvom *cache*-a od 64 kilobajta ukoliko želite da „džina" opremite sa više keša, tu su podnožja u koja možete utaći 128 ili 256 kilobajta 15 nanosekundnog SRAM-a. Preostalo prazno podnožje namenjeno je onima koje brzina *Intelovog* 80486 (koji u sebi sadrži i aritmetički koprocesor) ne zadovoljava – u njega možete utaći *Weitek 4167*

U računar koji smo testirali ugrađeno je 8 megabajta RAM-a u SIMM modulima ploča je po pitanju memorije izuzetno fleksibilna i na nju zavisno od tipa korišćenih modula, možete da ugradite 1, 2 4 5, 6 8 9, 10, 12, 16, 20 24, 32, 40 ili čak 64 megabajta RAM-a, naravno, pri izboru SIMM-ova treba biti pažljiv pošto se ne može sa svake konfiguracije preći na svaku – u konfiguraciju koju smo imali se na primer, moglo dodati još 4 ili 8 megabajta ali bi za proširenje na 32 megabajta trebalo zameniti i ugrađene 1 M SIMM-ove za one od 4 M Sve to je lepo opisano u okviru dokumentacije dobijene uz ploču.

Verovatno zbog činjenice da se sva potrebna memorija ugrađuje na osnovnu ploču nije predviđena mogućnost priključenja 32-bitne ekspanziona kartice – možete da utaknete šest AT i dve XT kartice što je u praksi više nego dovoljno, na računaru koji smo testirali bila su zauzeta dva mesta (disk kontroler i VGA) a mi smo treće popunili RS232/Centronika karticom Opšti utisak o elektronici je povoljan, ploča izgleda veoma solidno ako se izuzme jedna naknadno dodata žica koja se prostire njenom pozadinom, uz ploču smo dobili i papir proizvođača iz koga smo saznali da je puno ime ploče MB-C48650SI/P i da krivac za žicu nisu oni nego *Intel* koji je u (takođe priloženom) biltenu opisao nešto što se zove *Oscillator Lockup Problem* i što se povremeno ispoljava pri uključenju ploča zasnovanih na brzim verzijama 80486, rešava se postavljanjem tranzistora između oscilatora i napajanja za šta je očito poslužila pomenuta žica No bez obzira ko je kriv ploča deluje veoma solidno i tokom čitavog perioda testiranja nije pokazala nikakve probleme

Ugrađeni BIOS je AMI-jev produkt najnovije generacije Osim standardnog podešavanja datuma, vremena, tipa priključenih diskova i video adaptera, omogućava testiranje i formatiranje hard diska kao i programiranje specijalizovanih čipova na ploči u cilju



Server izuzetne brzine: Šutlić SC4050

poboljšanja performansi sistema. Za razliku od nekih ranijih verzija ovaj BIOS ne zahteva kucanje kriptičnih sadržaja pojedinih registara specijalizovanih čipova već odgovor na niz pitanja. da li želite eksterni ili interni procesorski *cache*, koje BIOS-e želite da keširate, kako se kešira video memorija, na koliko megaherca radi magistrala (možete da izaberete trećinu, četvrtinu petinu ili šestinu osnovne frekvencije, mi smo radili pod „punim gasom") i tome slično, ukoliko ne želite keširanje BIOS-a i grafike, čuvena 384 kilobajta se „vraćaju" u zonu *extended* memorije ali je tada, mora se priznati, rad primetno sporiji BIOS najzad obuhvata i par opcija vezanih za bezbednost sistema

Što se performansi tiče 486 ploča postiže rezultate koje smo od nje i očekivali – kao što se vidi iz tabele CPU komponenta RI testa završena je za 13 sekundi što je gotovo dvostruko bolje od do skora „standardno jake" 80386/33 MHz/64 K *cache* konfiguracije i tri-četiri puta brže od raznih 20-25 MHz 386 računara, poređenja sa SX i 286 pločama teško da ima ikakvog smisla Slične rezultate pokazuju i drugi testovi – *Norton*-ov SI (6.0) na primer daje 108.1 što je 3 puta bolje nego *Compaq Deskpro 386/33* (34.7) 25 puta bolje od standardnog AT-a i, ako je to bitno, 108 puta bolje od XT-a

ŠTA MOŽE SCSI

Najzanimljivija komponente sistema je, svakako, SCSI disk kontroler sa 4 megabajta keša i hard diskom od 210 megabajta Većina domaćih PC-ja nekada je bila opremljena MFM i RLL kontrolerima a u poslednje vreme dominiraju AT BUS (IDE) diskovi, ESDI su rezervisani za neka specijalna diskove velikog kapaciteta (npr WREN), dok je SCSI standard toliko malo zastupljen da ćemo uzeti slobodu da kažemo kako se ta skraćenica izgovara – „skazi" Pa ipak radi se o veoma rasprostranjenom standardu kako na PC tako i na većim sistemima – mnogi VAX-ovi i radne stanice veoma uspešno koriste SCSI diskove Tako široka primena svakako je prednost SCSI-ja a za korisnike PC-ja mnogo značajnija prednost je mogućnost povezivanja većeg broja diskova na isti sistem Ukoliko naime, koristite MFM RLL, IDE ili ESDI interfejs, u računaru možete imati najviše 2 hard diska, ponekad se mogu priključiti dva kontrolera (dakle četiri diska) ali se tu pojavljuju najrazličitije nekompatibilnosti, brojni problemi i glavobolje tako da uspešno priključenje više od dva diska na PC predstavlja pre izuzetak nego pravilo Ukoliko se opredelite za SCSI interfejs, problemi tog tipa su u startu rešeni – na isti računar možete bez problema priključiti i svih sedam diskova, a umesto nekog od njih i strimer traku CD ROM ili drugi neki drugi medij masovne memorije Da bi stvar bila još lepša, SCSI interfejs obično „sarađuje" sa IDE karticom tako da se u isti računar obično mogu priključiti dva AT BUS i jedna serija od 8 SCSI diskova – svakako više nego što će vam ikada zatrebati

A problemi? Nije da i njih nema – autor ovoga teksta je u prvim AT danima imao svakodnevne kontakte sa jednim SCSI diskom od 80 megabajta koji ga nije baš oduševio Pre svega, pri konfigurisanju interfejsa trebalo je „slagati" računar da disk uopšte nije instaliran to je rezultiralo znatno sporijim butovanjem i odbijanjem raznih uslužnih programa (*Norton Utilities* itd) da prepoznaju disk Najzad, performanse nisu bile na nivou koji smo očekivali od SCSI kontrolera tako da nam je ovaj interfejs nekako ostao u lošoj uspomeni. Ovo je bila dobra prilika da tu uspomenu revidiramo

Kontroler kalifornijske firme *Promise Technology* se punim imenom i prezimenom zove *DC 2040 ISA/SCSI Caching Disk Controller* i smešten je na jednu šesnaestobitnu AT karticu dobar deo te kartice zauzet je podnožjima za SIMM-ove, njihovim kombinovanjem može se obezbediti između 256 kilobajta i 16 megabajta keš memorije Na sistemu koji smo testirali bilo je ugrađeno 4 megabajta što je verovatno maksimum keš memorije koji ima smisla ugraditi U okviru prilično detaljne dokumentacije koju smo dobili uz kontroler opisana je ne samo njegova instalacija nego i moguća saradnja sa drugim IDE ili SCSI kontrolerom koji (eventualno) odlučite da instalirate Da bi DOS mogao da prepozna sekundarni SCSI kontroler u CONFIG.SYS treba ugraditi poziv PTIDOS.SYS drajvera za druge operativne sisteme (npr *Unix*) treba nabaviti odgovarajuću alatku koju firma *Promise Technology* obećava Osim tog drajvera na uslužnoj disketi je i nekoliko programa koji obezbeđuju proveru ispravnosti kontrolera, particionisanje diska uključenje, isključenje i „pražnjenje" *cache*-a i par sličnih operacija Svi programi su veoma jednostavni za korišćenje i solidno dokumentovani

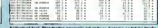
Kontroler sadrži i 8 kilobajta BIOS-a koji može da se postavi na bilo koju adresu (deljivu sa 2000H) između C8000H i DE000H BIOS se koristi samo u fazi inicijalizacije sistema kada (prema ID broju) određuje koji je SCSI disk primarni i postavlja pitanje treba li uključiti keš ili ne, pri normalnom radu se na

Slika 1



RI test SC4050 sa ili bez keša uporeden sa referentnim računarima

Slika 2



Aplikativni test. Standardne operacije pri radu sa bazama podataka izvršena na dve referentne mašine i SC4050 sa različitim varijantama keširanja

poziva pa će ga razne *Stealth* opcija QEMM-a i 386MAX a „prekriti" *high* memorijom

Jedna od lepih osobina SCSI standarda svakako je fleksibilnost. Priključeni diskovi se mogu razlikovati po tipu i kapacitetu mogu biti interni i eksterni – jedino što je potrebno jeste da na poslednjem internom odnosno eksternom disku bude po jedan tzv *resistor terminator pack* ako nema internih odnosno eksternih diskova oba potrebna RTP-a su na samom kontroleru Ukoliko posedujete diskove koji prevazilaze DOS-ove granice (504 megabajta – SCSI kontroleri podržavaju diskove kapaciteta 2 gigabajta), možete ih pomoću samog kontrolera podeliti na delove željene veličine – računar će imati utisak da pristupa većem broju manjih diskova dok je fizički sve to jedan ogroman disk U svakom slučaju računaru se u okviru SETUP-a saopštava da je priključen disk tipa 1 (oko 10 megabajta) a sam kontroler mapira prostor prema stvarnim karakteristikama diska.

HARDVERSKI...

Najzanimljivije pitanje se naravno odnosi na performanse, pri čemu pri proceni treba biti veoma pažljiv – sa jedne strane, računare obično testiramo „onakve kakvi su" dakle bez ikakvih softverskih alatki kao što je *cache* Na ovom sistemu *cache* je međutim, praktično ugrađen u hardver ali to ipak na znači da poređenja sa nekim drugim sistemom bez keša može da bude fer Zato smo se opredelili za nekoliko testova čiji su rezultati prikazani na slikama

Kad osnovu za testiranje koristili smo *data base* test koji smo svojevremeno napisali za potrebe *Unitest*-a Program smo najpre izvršili na konfiguraciji sa uključenim kešom (4 megabajta) i uporedili ih sa rezultatima dobijenim na 386/33 računaru sa sasvim sličnim AT BUS diskom (takođe *Maxtor*) Kao što se vidi poređenja praktično i nema – SC4050 je zavisno od operacije koju obavlja 3 10 puta brži Zatim smo koristeći jedan od uslužnih programa sa sistemske diskete isključili *cache* – kao što se i moglo očekivati, vremena su značajno porasla ali su i dalje primetno bolja od „referentnih" – u nekim slučajevima (npr računanje i sortiranje) velike razlika su svakako rezultat prednosti mikroprocesora, ali testovi pretraživanja indeksiranog pretraživanja i (donekle) indeksiranja pokazuju da korišćeni SCSI kontroler, čak i kada se keš isključi obezbeđuje otprilike 20-30% brži rad (na realnim aplikacijama!) u odnosu na široko rasprostranjeni IDE Razlika, sve u svemu, nikako nije zanemarljiva! Korišćenje *Norton*-ovog SI testa u prisustvu keša daje totalno „neuračunljive" rezultate koji toliko variraju da ih nema smisla izlagati test sa isključenim kešom daje 3 4 što je znatno manje od *Compaq*-a 386/33 koji daje 8 4 rekli bi smo da ova razlika predstavlja neku nesavršenost SI-ja pošto na aplikativnim testovima *Compaq* ni slučajno ne pokazuje tako visoke rezultate Starije verzije ostalih programa za testiranje diska (a i računara) poput *Speed*-a, *Coretest*-a i sličnih se jednostavno ne može koristiti jer nisu predviđeni za ovako „kosmičke" brzine – uglavnom se javi poruka o grešci zbog deljenja nulom

... I SOFTVERSKI KEŠ

Najviše nas je međutim, zanimala treća serija testova u okviru koja smo isključili hardverski keš a onda instalirali „običan" *cache* program (*PC Kwik* 4 02) dodelivši mu 4 megabajta XMS memorije Bez dovoljno prethodnog razmišljanja, nekako smo smatrali da će hardverski keš dati bolje rezultate među tim, *PC Kwik* je za 23% brže pretraživao za 13 posto brže računao pri indeksiranju su bili približno jednaki dok je hardverski keš „pobedio" kod indeksiranog pretraživanja (10%) i sortiranja (4%) Kada malo bolje razmislite ovi rezultati i nisu tako neočekivani kada se kešira *PC Kwik*-om, čitava komunikacija je usmerena na memoriju pa je izuzetno brza, pored ostalog i zahvaljujući memorijskom kešu Uz hardverski keš međutim podatak se traži preko ISA magistrale koja je davno zastarela – koliko god je disk kontroler brz, komunikacija i „dogovaranje" sa njim odnese određeno vreme Kada jednom pređemo na EISA *Micro Channel* ili neki treći moderno dizajniran *bus* kontroleri sa hardverski rešenim keširanjem će svakako „doći na svoje" do tada se postavlja vrlo ozbiljno pitanje vredi li investirati u takvu memoriju ili je bolje za toliko proširiti osnovni RAM pa onda njegov deo koristiti kao *cache* Razloga ima i za jednu i za drugu odluku – ako se opredelite za hardverski keš imate ga u svakoj aplikaciji i na svakom operativnom sistemu bez potrebe da investirate u keš programe i mučite muke sa njihovom kompatibilnošću sa QEMM-om *Windows*-ima OS/2 i drugim programima Ukoliko sa druge strane, proširite osnovnu memoriju dobijate keš nešto boljih performansi i što je svakako važnije mogućnost da tu memoriju, ako zatreba, „oduzmete" kešu i „pozajmite" nekom drugom programu kome je potrebna. Odluka, svakako nije jednostavna i u najvećoj meri zavisi od posla kojim se bavite

Na slici su prikazani i rezultati testa sa jednim megabajtom keša, što je svakako količina koju većina naših čitalaca koristi podatak dajemo radi poređenja. Nećemo trošiti prostor na komentarisanje svih rezultata – upada u oči da je indeksirano pretraživanje za nijansu brže obavljeno sa manjim kešom (??), što je verovatno posledica neke „rezonancije" između baze podataka i korišćenog indeksa procena performansi sistema nikada nije ni jednostavna ni „pravolinijska"

UBRZANI VIDEO

U računar je ugrađena *Camera Graphic Accelerator VGA* kartica kojom dominira 86C911 *GUI Accelerator* i u koju je ugrađen megabajt DRAM a što dopunjava memoriju ugrađenu u računar do 13 megabajta (8 na ploči 4 na kontroleru i 1 na kartici) U saradnji sa odgovarajućim monitorom kartica omogućava VGA i Super VGA modove 640•480 u 65536 boja 1024•768 u 256 boja kao i 1280•960 i 1280•1024 u 16 boja Uz računar smo dobili *Samtron* monitor koji obezbeđuje frekvenciju horizontalnog skaniranja 31 5 35 2 ili 35 5 kHz i frekvenciju vertikalnog skaniranja 50-90 Hz uz propusni opseg 40 MHz (-3 dB) – mogli smo dakle da idemo „samo" do rezolucija 1024•768 u *interlaced* režimu No slikom smo vrlo zadovoljni – bila je čak i stabilnija nego što smo očekivali za taj režim, uz jasne boje dobar kontrast i (barem prema specifikaciji) prihvatljivo zračenje Uz karticu smo dobili i tri diskete sa raznoraznim drajverima tako da smo bez problema instalirali *Windows 3 1*, *WordPerfect* i *Venturu* a priloženo je i sve što je potrebno korisnicima *AutoCAD*-a i par drugih popularnih programa Naročito bismo pohvalili drajver za *Windows 3 0* koji obuhvata i definicije potrebnih fontova – za razliku od mnogih konkurentskih kartica, *Graphics Accelerator VGA* vas ne tera da koristite obične *Windows* fontove u nestandardnim rezolucijama.

A brzina? I pored imena kartice, RI test je pokazao slabije rezultate nego na *Trident* VGA, ali ne zaboravite da RI testira isključivo tekst mod u kome se sa *Trident*-om zaista nije lako takmičiti Glavni specijalitet kartice je GUI prilagođen *Windows*-u i *Presentation Manager*-u, koji značajno ubrzava iscrtavanje linija, pravougaonika (*bitblt*) i popunjavanje pravougaonih površina uz potpunu VESA kompatibilnost U praksi sve to izgleda veoma dobro – *Windows*-i su primetno brži nego na pomenutoj *Trident* kartici Pokušali smo tu „primetnost" da transformišemo u brojke startujući jedan opsežan demo program pisan u *Turbo Pascal*-u *for Windows* – na 486/33/64 K računaru sa *Trident* karticom crtanje je završeno za 78, a na SC4050 za svega 55 sekundi, tih 30% se itekako primećuju naročito kada se radi sa *CorelDraw*-om ili nekim sličnim „vektorskim" programom

GOTOVO BEZ PROBLEMA

Na SC4050 smo isprobali raznorazne programe, počevši od DOS-a i *Windows*-a preko raznih *DOS Extender*-a do *Novell*-a i OS/2. Problema praktično i nije bilo – i QEMM i 386MAX savršeno upravljaju memorijom, *Windows* i *PC Kwik* se (po potrebi) slažu *WordPerfect* savršeno radi i u proširenim tekst modovima *Novell* daje indeks brzine 1380 (najbrže što smo do sada videli) čak i „najprljavije" igre poput *Crime Way*-a su potpuno sarađivale sa VGA karticom Jedini problem koji smo imali doneo nam je OS/2 2.0 koji baš nikako nije hteo da se instalira – posle par uvodnih ekrana pojavljivala se poruka o internoj grešci, sadržaji registara i predlog da se javimo dileru koji nam je prodao softver Već smo pomislili da neće biti prilike da se OS/2 isproba na ovako pogodnoj platformi, kada nam je u trenutku inspiracije „sinulo" da isključimo keš (to se, srećom, može obaviti i tokom butovanja računara) i pravo čudo instalacija je prošla a OS/2 evo već nedelju dana sasvim solidno funkcioniše s tim što moramo dobro paziti da pre njegovog startovanja isključimo keš na kontroleru To je svakako pre nesavršenost OS/2 2.0 (očito još nije prilagođen specifičnijim hardverskim konfiguracijama) nego ovog računara i njegovog kontrolera ali problem ostaje

Sve u svemu utisci o SC4050 su veoma povoljni, svakako najmoćnija PC mašina sa kojom smo imali prilike da se detaljno upoznamo Obzirom na performanse i cenu, svakako će naći primenu, pre svega kao server lokalne mreže pod *Novell*-om ili *Unix*-om kada će keš na kontroleru doći do izuzetnog izražaja (u takvim slučajevima, naročito ako je server posvećen jedino održavanju mreže (*dedicated*) vredi razmisliti o monohromatskoj varijanti), kao grafička radna stanica izuzetnih performansi ili procesorska stanica za korisnika velikih potreba. Uzevši u obzir (za ovakav računar) umerenu cenu, poslovnost preduzeća „Šutlić" (platite i odmah nosite računar) i dobre uslove servisiranja, računar vam toplo preporučujemo naravno ukoliko vaše potrebe opravdavaju takvu investiciju

Korisna adresa

Šutlić
11000 Beograd Kosovska 32/I
Tel (011) 332-697 Fax (011) 345 126

VI ZNATE BOLJE

Recimo da imate AT, ili ko zna, možda i XT, a niste očarani postojećom ponudom gotovih mašina. Šta da radite – da očajavate ili nešto da preduzmete? Pa da preduzmete, naravno, a evo šta, kako, kod koga i pošto.

Dejan V. Veselinović

Pretpostavimo da imate neku sada već zastarelu mašinu koju želite da osavremenite ili da ste rešili da zasučete rukave i samostalno sklopite sebi računar Tu naravno postoji niz pitanja, iako ni slučajno ne pledirano na to da je naš pristup jedini ispravan, ipak vam ga podnosimo na uvid makar kao neki opšti stav kako bi to trebalo uraditi

OSNOVNA KONCEPCIJA

Ona se uvek sastoji od tri stvari šta imate šta vam treba ili šta želite i koliki vam je budžet Oni koji tek počinju ne moraju odgovarati na prvo pitanje a oni koji nešto već imaju treba dobro da se zamisle nad onim što već imaju pre svega u smislu ima li razloga ponešto i zadržati, ili je ipak bolje prodati sve i početi iznova

Oko zadržavanja delova ili prodaje rekli bismo vam samo jednu stvar po našem iskustvu najbolje je prodati postojeće i početi od početka Primer imate RLL disk od 32 MB ili ste „lupali fiz" disku „Seagate" ST 251 pa sada radi kao RLL disk sa 63 MB Ta tehnologija je definitivno mrtva, pa ako vam ne daj bože nešto otkaže veoma teško ćete nabaviti rezervne delove Zato predupredite neminovnost i lepo prodajte ono što imate a pare iskoristite za novu opremu

Sa praktične strane podsećamo da samo *Windows* zauzima dobrih 7 MB prostora a svi bolji tekst procesori još toliko Dakle samo za to operativni sistem i par korisničkih programa vam treba nekih 17 20 MB A ako se opredelite za OS/2 V 2. on sam će pojesti nekih 30 MB

Imajući sve to u vidu, kao i sve veće zahteve programa i najzad smatrajući da se unapređenje ne radi svaki dan a ne bi trebalo ni svake godine, sali smo i malo se preslaudumili Ako se opredelimo za SX moći će sve što nam treba, ali neće biti baš preterano brzo a *Windows* ionako nije mnogo brz Dakle, ne bi trebalo da idemo na baš samo dno, na ulazni nivo nego ipak za jedan korak iznad

Da bismo OS/2 uopšte mogli da pokrenemo treba nam minimum od 4 MB sistemske memorije to je dakle apsolutni minimum ispod koga ne želimo da se pomeramo Sličnom logikom ako OS/2 zahteva najmanje 30 MB prostora i ako uz njega na disk stavimo jedan bolji tekst procesor program za komunikacije, par korisničkih programa kao recimo *PC Tools* i neki recimo grafički program, potrošićemo između 48 i 53 MB prostora na disku Smatramo da je mudro da početna količina ne prelazi više od trećine kapaciteta diska a to znači da tvrdi disk mora imati najmanje 100 MB

Najzad sve ispod EVGA (800×600) grafike danas zaista nema više bilo kakvog smisla sve se piše samo za kolor pa ukoliko se ne bavite baš čisto tekstualnim bazama podataka ili računovodstvom trebaće vam i neki pristojan video podsistem

E da, kažete vi, ali sve to zajedno košta đavo i po, a za te pare ja bih da znam šta kupujem Pošteno – a zato se i radilo na ovom članku Naša ideja je bila da cenu ni u jednom momentu ne izgubimo iz vida ali ni da ne podredimo sma baš sve samo ceni Jeftino danas može lako postati veoma skupo sutra kada za nekoliko meseci shvatite da ste pogrešili ovde ili onde pa onda morate da prodajete tada već polovnu robu da biste nekako skrpili za novo

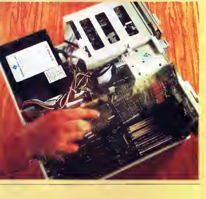
Dakle, radi se o stavu da nekakva optimizacija prethodi ceni a da ipak tu i tamo treba napraviti kompromis radi cene Ali i taj kompromis treba da bude takav da vam poboljšanje što lakše padne a ne da sutra opet grcate

KOMPONENTA PO KOMPONENTA

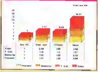
Srce računara je matična ploča pa smo od nje i počeli Prvo smo sabrali sve rezultate koje smo do sada sakupili na raznim testovima a nakon toga smo proveli dobra dva dana sravnjujući ih, pošto su bili rađeni pod raznim verzijama našeg „Unitesta" Tako smo dobili tzv istorijski prosek odnosno indeks prosečnih performansi svih mašina u toj kategoriji A kategorija je procesor 80386 (AMD ili „Intel" svejedno) na 33 MHz ova platforma se na većini spiskova prodavaca danas nalazi odmah iznad procesora 386SX na 25 MHz odnosno predstavlja prvi naredni korak Naravno sa 4 MB sistemske memorije

Probali smo nekoliko matičnih ploča U celini gledano sve se nalaze iznad istorijskog proseka što sasvim lepo svedoči o sazrevanju tehnologije Ogromna većina tih ploča spada u tzv univerzalne generičke proizvode odnosno ploče koje možete uz male izmene pretvoriti u bilo šta Prostim ubacivanjem odgovarajućeg procesora i uz par preklopnika, možete imati sve od 80386 na 25 MHz sa ili bez keša sve do 80486 na 50 MHz sa 256 KB keša i to sa procesorom koji radi sa dvostrukim taktom Ovo je sasvim razuman potez sa aspekta proizvođača ali ima i svoju cenu a sa aspekta kupca nema mnogo smisla uprkos reklami Baš da vidimo ko će menjati samo procesor koji košta 3/4 cene cele ploče posebno kada ustanovi da će morati da zameni i skup SRAM koji služi kao keš čime će faktički platiti 5/6 cene nove ploče uz rizike koji slede pri takvim zamenama

Da skratimo – nama se najviše dopala ploča koju prodaje firma „Comtred" (i njeni dileri) a koja koristi drugu generaciju logičkih čipova firme Chips & Technologies" Umesto sedam malih čipova plus „Intel" 82385 keš kontroler sada na ploči ima ukupno tri VLSI čipa plus naravno procesor u našoj verziji od firme AMD Ploča je strogo namenska što znači da ne možete menjati bilo šta jer je čak i kristalni oscilator zalemljen za ploču Za uzvrat sa njom smo postigli najveću pozitivnu razliku u odnosu na do sada najbržu mašinu u ovoj klasi, mada se istini za volju ta razlika svodi na vrlo akademskih 1-6% Međutim ploča je retko „čisto" urađena, sa odličnom štampom, bez raznoraznih žičica a zaista jedina zamerka ko-



Slika 1



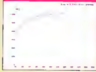
Mašinski testovi

Slika 2



Programski testovi

Slika 3



Tvrdi disk - brzina prenosa

ju imamo su dva kratkospojnika, koje bismo mi lepše zalemili (bili bi bar ravnije postavljeni!). BIOS je od firme AMI a ne dopada nam se to što sadrži čitav niz potpuno novih opcija, za koje opet (i posle svih ovih godina!) nema podataka o tome čemu služe. No, najbrža je pa makar i za nekoliko procenata.

Oko diska bi se već dalo dosta sporiti. Uprkos trenutnoj nepopularnosti firme „Seagate" mi imamo odlično iskustvo sa njihovim proizvodima potpuno isto kao i sa firmom „Quantum" (koja je naša stara slabost). Znamo da popularan stav nije ni slučajno ovakav, ali podsećamo da se u svetu proda kudikamo više „Seagate" diskova od svih drugih, pa je i statistički očekivano da se oni „najviše" kvare. To je kao kada kažete da je „Boeing 707" nepouzdan avion jer najviše pada, a zaboravite da njih ima u službi više od svih ostalih vazduhoplova zajedno.

Elem, probali smo nekoliko diskova u saradnji sa ovom pločom i na kraju se opredelili za „Seagate" model ST 3144A. Radi se o savremeno koncipiranom disku formatiranog kapaciteta od 130 MB, što bi za većinu korisnika trebalo da bude sasvim dosta. Interesantno je da smo sa njim postigli rezultate saradnje sa pločom koji su bili u samom vrhu uopšte i to sa potpuno prosečnim kontrolerom, jeftinim i anonimnim.

Sa video kartom smo se već malo pomučili, jer je izbor zaista veliki. Odlučili smo da baš na njoj napravimo kompromis na ceni i o užasa nad užasima, upotrebismo „Oak Technology" OTI067 video karticu. Ona standardno ima 256 KB memorije, ali je proširiva do 512 KB čime postaje moguće imati rezoluciju do 800×600 sa 256 boja i 1024×768 sa 16 boja. Pošto smo na budžetu, pretpostavili smo da veliki broj boja u svemu iznad rezolucije od 800×600 više nema značaja, pa smo ovu karticu odabrali na osnovu cene. Pošto „Trident" kartice hvala na pitanju - ne trošimo ni pod razno (usled zaista enormnih varijacija od serije do serije), a „Tseng" i WD su već više nego duplo skuplji i nismo imali baš neki izbor. Kartica je solidna, ali ništa jače od toga, no zato je jeftina. Najzad, pretpostavili smo crno-beli VGA monitor, a oni najčešće ne idu dalje od VGA rezolucije, odnosno 640×480 (ekonomija!).

Ostatak je strogo po želji. I/O kartica vam ne gina ako je već nemate, a isto važi i za AT Bus (IDE) adapter. Tastatura je po izboru, kao i kućište i monitor.

DOKTORE, KAKVA JE DIJAGNOZA?

Pa radi, naravno, radilo je od početka i radi i sada, a nismo ga mučenika gasili evo već 24 dan (576 radnih časova). Kako radi? Lepo, sasvim lepo.

Na našim uobičajenim tabelama prikazujemo četiri vrednosti. Prva kolona se odnosi na našu apsolutnu referencu, jednu SX mašinu na 25 MHz. Treća kolona prikazuje rezultate dosad najbrže mašine sa kojom smo ikada imali posla (Gama 486 na 50 MHz), a poslednja kolona prikazuje istorijski prosek performansi. Samim tim možete tačno videti koliko je ovaj naš predlog brži od reference, sporiji od najbržeg na tržištu i kako stoji u odnosu na prosek ovakvih mašina (u kontekstu matične ploče). Napominjemo da smo rezultate za istorijski prosek morali da preračunavamo radi ujednačenja, a to uvek podrazumeva i greške, za koje tvrdimo da se nalaze u opsegu ± 1.25 %.

Kao što se vidi, u hardverskim testovima naš predlog je brži od istorijskog proseka za 8%, a od najbrže ovakve mašine do sada za skromnijih 2%. Od apsolutne reference je brži solidnih 67%, a podsećamo da je referenca dobro opremljena („Quantum" LPS disk, takt 25 MHz, četvorostruko preplitanje). Na programskim testovima, ovaj indeks se neznatno spustio padajući na 1.57. Međutim, to je ipak veoma osetnih 57% brže od reference, što se i te kako oseti pod prstima. Sa druge strane, to jedva nešto malo brže od trećine brzine mašine sa procesorom 80486 na 50 MHz, ali treba znati i to da glavne razlike u brzini potiču od ugrađenog matematičkog koprocesora. Primera radi, „Gama" je 2.28 puta brža u obradi teksta, ali je 4.5 puta brža u CAD-u. Razlike u testovima pod *Windows*-om potiču između ostalog i od inherentno brže video karte.

Konkretno, brzina protoka na ovoj video kartici je 1.8/1.5 MB u sekundi, a WD 90C30 ima brzinu propusta od 3.5/2.7 MB u sekundi. Dakle, ako bismo ugradili tu karticu, dobili bismo bezmalo duplo bolje performanse na video, i ne samo na papiru. Naime, mi smo broj stanja čekanja na sabirnici podesili na 2 (propust 2.5/2.2 MB/s), a ako bismo broj stanja čekanja smanjili na 1 (što radi i ovako, ali nema smisla), dobili bismo brzinu od 3.0/2.5 MB/s, opet ispod kapaciteta kartice, ali bar 65% brži video. Uzgred, prva brojka se odnosi na čitanje, a druga na pisanje posredstvom sabirnice.

ŠTA KAŽE Dr. FRANKENŠTAJN?

Sa 4 MB sistemske memorije nije teško odreći se DOS-a 5.0 i koristiti DOS 3.31. Na njega nije problem natovariti prvo *PC Kwik* program za keširanje diska, a završiti sve sa nekim *QEMM*-om ili *Maxom 6.0*. Na kraju čoveku ostane u bajt isto koliko i da je radio sa DOS-om 5.0, sa dodatnom prednosti da 2.5 MB mogu po potrebi biti produžena ili proširena memorija, što koji DOS 5.0 na ume da izvede.

A kada dodate 1 MB keša, mašina naprosto poleti. Indeks na testu baze podataka sa 1.25 skoči na 3.39, a ni ostali indeksi ne ostaju isti. Na neki način tek tada oslobodite sve potencijale ploče, pa još ako umesto Oak kartice ubacite WD, onda zaista uzlećete. U poređenju sa recimo nekom AT mašinom na 16 MHz, razlika je otprilike kao kada ste sa PC-a prelazili na AT.

KELNER, RAČUN MOLIM!

Naravno, ostaje pitanje koliko sve to košta. To je pitanje na koje je veoma teško odgovoriti, jer smo sada u situaciji da moramo da postavljamo dva pitanja: Koliko to košta u DEM i pošto je DEM kod vas? Ovakve matične ploče u inostranstvu tipično koštaju 550-650 DEM, kako koja i kako gde, mi bismo vam savetovali da ipak prvo probate kod „Comtrade" ili nekog od njihovih dilera. Kada se već kod njih raspitujete, pitajte i to kako stoji stvar sa onim prelepim, ta Hi-Fi crnim kućištima, sa odgovarajućim monitorima i tastaturama. Bilo bi lepo imati ih u kući – ali mi, usled pomanjkanja prostora, računar držimo ispod stola. Na cenu ploče dodajte oko 280 DEM za 4 MB memorije u obliku SIM modula.

Disk možete kupiti kod mnogih prodavaca, a spada u jeftinije diskove na tržištu. Cene u inostranstvu se kreću oko 725-775 DEM, u zavisnosti od izvora. Kod nas tipično košta oko 1.050-1.100 DEM, ili oko 8.5 DEM po megabajtu, što je dobra cena. Još kada znate da mu je prosek brzine prenosa za blokove od 1 do 64 KB preko 1.100 KB/s, može se reći da je disk dobar. Ako biste odmah dodali gas do daske, kupite za nekih 1.300-1.400 DEM (u inostranstvu) disk sa oznakom „LPS" firme „Quantum" – formatiranog kapaciteta od 245 MB, potom kod kuće bacite „Stacker".

Ako ste na tesnom budžetu, uzmite opisanu „Oak" karticu sa nekim crno-belim VGA monitorom. U inostranstvu to će vas koštati oko 330 DEM (monitor i karta sa 512 KB), a kod nas oko 70% skuplje. Međutim, ako ikako možete, umesto opisane kartice uzmite onu sa čipom WD 90C30 – to je najbrža što se trenutno prodaje za neke razumne pare – i dodaće vam na cenu nekih 130-150 DEM.

Dakle, ploča plus disk plus video plus memorija plus kućište plus sve ostalo izađe oko DEM 2.200 u inostranstvu (za VGA mono). To nije malo, ali nije ni preterano za ono što dobijate.

Ako kupujete sve odjednom, možda bi vas se isplatilo da se malo raspitate kod nas za baš ove komponente ili eventualno neke malo odlje. Ako pak želite da zamenite postojeće, krenite od matične ploče, preko diska do video podsistema, koliko vas već budžet bude nosio.

I za kraj, reč upozorenja – nemojte se opredeliti za samogradnju ako nemate iskustva, ili ako ne možete da regrutujete nekoga ko to dobro zna. Ipak se tu radi o osetljivim stvarima koje nisu za neiskusne. Ali, ako možete i umete...

IZ KAPI MASTILA

Rekio bi se da je tržište štampača potpuno polarizovano između matričnih, s jedne, i laserskih printera, s druge strane. „Canon" i HP su, međutim, nastavili da razvijaju i druge tehnologije. Da pogledamo dokle su stigli.

O tehnologiji matričnih ili udarnih (impaktnih) štampača se već veoma mnogo zna. Od prvobitnih devet, ovi štampači su stigli do 24 iglice, a bilo je pokušaja i sa 48 iglica. Njih je, međutim, onemogućio brz razvoj laserske tehnologije koja je u međuvremenu dovoljno pojeftinila i dovoljno se razvila da je sve preko 24 iglice učinila besmislenim.

U celom tom procesu odvijala su se dva paralelna toka. S jedne strane, tehnologije je stalno pojeftinjevale, što je normalno usled sve većih serija, a s druge strane nužno se nastavilo sa razvojem alternativnih tehnologija. Razloga za drugo ima nekoliko: prvo, uprkos velikom stepenu smanjenja matričnih štampača, kao recimo u slučaju firme „Diconix" (vlasništvo „Kodaka") ipak su mali matrični štampači ostali zakinuti za kvalitet otiska; drugo, smanjenje dimenzija ne mora nužno rezultirati u smanjenju kvaliteta; treće, segment korisnika koji je zainteresovan za fizički male štampače je isto tako zainteresovan i za odličan kvalitet otiska i četvrto, klasične tehnologije se baš nisu proslavile u štampanju u bojama, što znači da bi prelazak na boju verovatno podrazumevao drugačije principe rada.

Od drugih tehnologija podsetićemo na dve koje su najviše obećavale. Prva je bila štampanje pomoću termalnog prenosa, u čijoj osnovi je ležalo štampanje zagrevanjem. Ovo je bio veoma tih način štampanja koji je potencijalno nudio veoma velike rezolucije na veoma malom prostoru. U osnovi nije uspeo zbog relativno velike cene proizvodnje i zbog potrebe za specijalnim trakama i, što je još gore, specijalnim papirom. Sve zajedno, to je davalo nepraktične i skupe štampače.

Druga, svojevremeno popularna tehnologija je koristila male cevčice kroz koje se mastilo izbacivalo na papir. Ovu tehnologiju je posebno negovala američka firma „Hewlett-Packard", uvek u saradnji sa japanskom firmom „Canon", koja je taktički proizvodila sve delove. Kao i termalna, i ova tehnologija je nudila male fizičke dimenzije, dobar mada ne i odličan kvalitet štampe – na žalost, uz relativno veliku cenu i uz stroge zahteve za posebnim vrstama papira. Za razliku od termalnog transfera, ova tehnologija nije nestala, već je doživela značajna evoluciona izmene.

Upravo ova tehnologija u novom izmenjenom izdanju čini osnov dva značajna proizvoda današnjice. Prvi je HP 500c, stoni štampač sposoban da radi u nizu boja po razumnoj ceni, a drugi je „Canon" model BJ-10e. Između njih postoji čitav niz razlika, od namene do cene, a zajedničko im je to što oba modela pravi ista firma – „Canon". Prvi model je inherentan HP-u, a drugi se, nalik ostalim proizvodima „Canona", prodaje u nizu izvedbi pod drugim imenima, kao recimo „Star", „Brother" itd.

LIČNI OPIS

Izuzetno skromne dimenzije štampača (vidi Tabelu 1) zaista oduševljavaju, iako nam nije prvi put da se susrećemo sa ovako malim modelima (ranije smo prikazali sličan model firme „Siemens/Mannesmann"). Iako je sav napravljen od plastike, što je bila nužnost imajući u vidu namenu, dno koje predstavlja neku vrstu samonoseće šasije je napravljeno od savijenog lima debelog bar ceo milimetar. Vođice glave su napravljene od metala, a sve ostalo je čista plastika.

Dejan V. Veselinović

Iznutre, lako ćete uočiti red od deset preklopnika pomoću kojih ćete konfigurisati ili rekonfigurisati štampač. Odmah iznad samih preklopnika, nalazi se detaljan opis namene svakog preklopnika, tako napravljen da vam ni priručnik ne treba za čitav posao (pod uslovom da znate značenje skraćenice kao što su „CR", „LF" i „AGM").

Na prednjoj strani se nalaze signalne diode i pet (po svemu sudeći) membranskih tastera. Signalne diode označavaju radno stanje, stanje ON LINE, odsustvo papira i grešku u radu, dakle, sve kao na nekom „velikom" štampaču. Tasteri su takođe prilično standardni, mada dva čine nekakav izuzetak. Prvi prebacuje štampač u stanje pripravnosti (ON LINE), drugi dodaje red (LF/FF) ili određuje lošiji odnosno kvalitetniji režim štampe, trećim se bira veličina slova, četvrti omogućava pomeranje unapred, peti pomeranje unazad ili čišćenje.

To čišćenje je interesantan proces. Štampač valja uključiti, a potom pritisnuti zajedno četvrti i peti taster (FWD ADJ i REV ADJ). Time ćete otpočeti proces čišćenja, koji se svodi na propuštanje malo mastila kroz cevčice ali bez pomeranja same glave. Teorija je da će mastilo izbaciti napolje svu zaostalu prljavštinu i papirnu prašinu. Taj proces bi trebalo da traje oko sekundu i po do dve, a prekida se ručno, pritiskom na ON LINE taster. Ovaj proces treba obaviti kada god primetite da slova ne izgledaju kako treba, odnosno da neka ili neke od cevčica ne funkcionišu kako treba. Simptom je slovo sa neravnim ivicama ili, u gorem slučaju, bled otisak.

Lična karta	
Vrsta proizvoda	A4 štampač
Proizvođač	Canon, Japan
Model	BJ-10e
Tehnologija otiska	Mlaz mehurića
Format papira	Do A4
Veličina ŠxVxD	310×47.5×216.5 mm
Težina	1.8 kg (bez baterije)
Napajanje	Spoljno 220–240 V 50 Hz
	Opciona NiCd baterija
Potrošnja struje	25 VA
Buka	Manje od 45 dB (a)
Međumemorija	3 kB (opciono 37 kB)
Emulacije	Canon BJ-130e
	IBM Proprinter
	IBM Proprinter X24/XL24
	IBM Graphics Printer
Štampanje	
– Teksta	Dvosmerno
– Grafike	Jednosmerno
– Brzina za tekst	83 znakova/s
– Brzina za grafiku	– –
– Grafičke rezolucije	do 360×360 tačaka/inču
– Maksimalna širina	do 203 mm/redu
– Ugrađeni fontovi	Courier
	veličine 10, 12, 17 i PS
– Maks. znakova u redu	do 136 efektivno
– Kodne strane	437, 850
– Niz znakova	IBM niz 1 i 2
Potrošni materijal	
– Vrsta papira	bankpost ob čni
– Papir	B5, A4
– Težina papira	52-105 g /m2
– Debljina lista	do 0.2 mm
– Traka/kaseta	Specijalna Canon
– Vek trajanja.	a) do godinu dana
	b) do 700 000 znakova
Prodavac	Intel
Adresa	Bul. Lenjina
Telefon	(011) 135–420
Fax	(011) 135–929

Druga egzotična funkcija štampača je sistem resetovanja ili vraćanja u početno stanje. Prvo morate istovremeno pritisnuti ON LINE i PITCH taster, a potom samo PITCH taster šest puta – ni manje ni više. Nismo perfekcionisti, ali ovo nam deluje baš uvrnuto: mnogo bi nam lakše bilo da štampač jednostavno isključimo i potom uključimo, iako to nije preporučljivo kao praksa (ali valjda se neće ni svaki čas zaglavljivati).

TEHNOLOGIJA ŠTAMPE

Princip rada ovog štampača se svodi na sledeći pojednostavljeni metod: mastilo se izbacuje kroz cevčice, kojih u glavi za štampanje ima 64. Mastilo se pod pritiskom ubacuje u cevčice, gde se pomoću električnih impulsa veoma brzo zagrejava i gotovo trenutno pretvara u mehuriće. Linijom manjeg otpora, ti mehurići stvaraju pritisak i izlaze iz cevčice, čime se u njoj za momenat stvara vakuum (usled nastupanja dela ciklusa bez grejanja), koji se brzo popunjava narednom kapljicom mastila.

Po „Canonu", ovaj princip ima nekoliko značajnih prednosti nad svojim pretečama. Prvo, konstrukcija cevčica i cele glave je značajno jednostavnija i pouzdanija, a time i jeftinija. Drugo, usled većeg broja cevčica nego u slučaju klasičnih matričnih štampača (64 umesto 24), kvalitet otiska je veći u ovom slučaju. I treće, pošto nema udaraca od strane udarnih iglica, ceo proces štampanja se odvija daleko tiše.

Jedini mogući problem koji se ovde primećuje je pitanje trajnosti glave za štampanje. No, upravo je to pojeftinjenje proizvodnje omogućilo „Canonu" da izvede svoj već dobro poznati trik sa laserskih štampača – glava (selenski valjak) je ugrađena u kasetu sa mastilom (ugljenom prašinom). Dakle, kao i kod lasera, zamenom medija za štampanje ujedno menjate i glavu. Sve je to montirano u jedan plastični blok koji se veoma lako i savršeno čisto zamenjuje za manje od jedne minute. Trik nije loš, nema šta.

BLIŽE LASERU

Odmah da priznamo – otisak je dobar. To dobar znači da je na nivou odmah iznad bilo kog boljeg matričnog štampača sa 24 iglice – a to nije ni malo lako postići, verujte. U svim standardnim režimima rada sa veličinom od 10 znakova po inču, ovaj štampač sa mirne duše može porediti sa matričnim, pa i sa laserom, jer se nalazi negde između te dve klase.

Ovo je proba
Ovo je proba Car
Ovo je proba Canon BJ-
Ovo je proba Canon BJ-10e
Ovo je proba Canon BJ-10e 12 cp
Ovo je proba Canon BJ-10e 17 cpi
Ovo je proba Canon BJ-10e 17 cpi Half Height
Ovo je proba Canon BJ-10e Prop

Kvalitet otiska

U slučaju većeg broja znakova po inču, što će reći kada koristite manja slova, događaju se dve međusobno suprotne stvari. Po svemu sudeći, štampač smanjuje slova smanjivanjem broja cevčica koje koristi, čime dolazi do efekta tanjih linija koje grade slova. Pored veličine, korisnik ta manja slova percipira kao subjektivno bleda, što nije mnogo zgodno. Kada u red strpate celih 136 znakova, slova postanu jako bleda, ali su neverovatno čitka i jasna: do sada nismo videli ni jedan matrični štampač koji čak i prilazi blizu ovom nivou čitkosti, a ni naš laser ne bi izvukao takav nivo čitkosti, mada bi imao veću debljinu slova (dakle, i bolji kontrast).

Suprotno tome, kada povećate slova, pa umesto 10 na jedan inč smestite samo pet znakova, dobićete slova sa kontrastom na nivou lasera (a boljeg nema), mada uz već lako uočljive izobličenja samih slova, tipski sasvim nalik na neravnine koje proizvode matrični štampači, premda su te neravnine manje izražene.

Naravno, sve ovo je sasvim logično i očekivano, ali fascinira kada to i vidite u praksi. Za razliku od traka koje koriste matrični štampači, ovaj model teorijski ima konstantan nivo kontrasta bez obzira na to da li je kaseta sa mastilom potpuno puna ili popunjena samo petinom mastila. To je bezuslovno prednost u odnosu na bilo koji matrični štampač. Sa druge strane, smanjenje slova smanjivanjem aktivne matrice nužno bledi samo slovo, a pošto u glavi ima 2,67 puta više otisnih elemenata, sasvim je normalno to što su mala slova jasnija na ovom nego na drugim štampačima. Ali, manja aktivna matrica znači i manje mastila, jer cevčice ne kompenzuju veličinu aktivne matrice (niti to mogu).

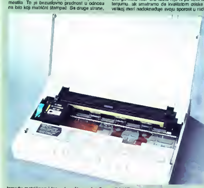
Što sa grafike tiče, tu stvari već stoje drugačije. Recimo, na testu sa nijansama sivog, mali „Canon" je na muku uspeo da izvuče četiri ne baš jasno prepoznatljive nijanse. Pride je sve to izveo sa veoma velikim zasićenjem papira mastilom, pa je mastilo čak malo probilo papir (mada ne strašno, papir je i dalje bio sasvim upotrebljiv).

U slučaju linijskih (čistih, bez nijansi) crteža, kvalitet otiska je zaista veoma dobar: po kvalitetu se nalazi negde u sredini između najboljih matričnih i laserskih štampača. Drugim rečima, laser nije, ali je bolje od bilo kog matričnog štampača koji smo mi do sada videli.

ŽIVOT SA BJ-10e

Jedini problem sa kojim se možete susresti u slučaju upotrebe ovog štampača je njegovo nepodržavanje „Epson" modela; sa druge strane, malo je ozbiljnih programskih paketa koji ne podržavaju bar neki od IBM štampača. Ako možete da birate, „Canon" preporučuje da se opredelite za IBM Proprinter XL24 kao najpoželjniju emulaciju. Mi smo ih poslušali i za sve probe i rad sa štampačem smo koristili baš ovu emulaciju (kada podelimo sa ocenom cene (koj „Canon" BJ-130e, koji je naveden kao alternativa).

Odsustvo buke tokom rada je zaista osvežavajuće, pa iako očekujete mir, ipak se prijatno iznenadite kada se vaša očekivanja ispune u većoj od očekivane mere. Zadovoljstvo je videti ga kako radi. Brzina nije ni po kom kriterijumu, ali smatramo da kvalitetom otiska u velikoj meri nadoknađuje svoju sporost u radu. Brzina je lepa stvar, ali kod štampača ipak nije sve ni samo u brzini.



Između matričnog i laserskog štampača: Canon BJ-10E

Više je nego pogodan za rad u svakom ambijentu. „Canon" ga je po svemu sudeći namenio pre svega vlasnicima prenosivih računara formata A4, što se i po priručniku vidi. Objektivno, više je nego pogodan za njih, pre svega zbog malih dimenzija, male težine i mogućnosti rada sa akumulatorom, ali je i više nego pogodan za rad kod kuće. Opet, usled malih dimenzija, možete ga staviti gde god hoćete i gotovo nikada vam neće smetati, a pošto je mali i lagan, bez po muke ga možete strpati i u neku fioku kada vam ne treba, ili ga čak i držati u njoj sve vreme.

Iako je sav od plastike, mora se priznati da je zaista dobro napravljen. Jedina stvar koja nam se ne dopada što je od plastike, ili tačnije, od ovakve plastike je ručka kojom se kaseta sa mastilom i glavom zaključava na mesto. To je i jedina ne baš svetla tačka ovog modela: ne možemo se požaliti da nam je loše radila, ali nežno sa njom, veoma nežno.

Neka rešenja nas baš i ne oduševljavaju, mada nemamo nekih izraženih zamerki: recimo, tasteri ne stoje baš onako čvrsto kako bi moglo i trebalo, komanda za prebacivanje sa listova A4 na koverte niti je pogodno smeštena, onako na dnu kućišta, niti je zgodna za rad, i tako dalje. Međutim, ni jedna od ovih zamerki nije krupnija, a istini za volju, neke od njih su u velikoj meri i stvar ukusa.

IZVAN SERIJE

Kućište i opštu obradu zaista nije moguće oceniti bilo čime sem najvećom ocenom, uprkos nekim zamerkama. Isto važi i za elektroniku. Apsolutne performanse koje se odnose na samu brzinu štampanja donose sedam poena: po brzini bilo bi i mnogo manje, ali ovakva tišina tokom rada zaslužuje nagradu. Zato relativne performanse donose devet poena, pre svega zbog izvanrednog kvaliteta otiska, a deseti je kazna za ipak spor rad. Garancija povlači pun broj poena. Time se dobija prosečna ocena od odličnih 9,2 (od mogućih 10).

Sada dolazimo do cene. Prodajna cena je 800 DEM ekvivalent u dinarima, što je izvanredna cena kako god je razmatrali. Rezervna kaseta sa mastilom košta 80 DEM, u poređenju sa oko 20 DEM za standardnu „Epson" traku, ali ovo mastilo traje oko 3,5 duže od tipične trake (700 000 prema oko 200 000 znakova za traku), mada je tačno i to da traku možete osvežiti, a ovu kasetu ne. Najzad, opet se nužno vraćamo na kvalitet otiska koji prevazilazi matrične štampače, pa kada sve to saberemo i oduzmemo, dobijamo ocenu od 8,1.

Kada ocenu kvaliteta podelimo sa ocenom cene (koja je, da podsetimo, negativna veličina – što manja ocena, to bolja), dobijamo indeks vrednosti od 1,17. Sve preko jedinice znači da se radi o odličnom poslu: da uređaj i više nego vredi svoju cenu.

To bi i bila naša konačna ocena ovog uređaja. „Canon" BJ-10e je pre svega namenjen ljudima koji moraju da nose štampač sa sobom (vlasnici prenosivih računara, trgovački putnici, poslovni ljudi itd), ali savršeno dobro zadovoljava i umerene potrebe za kvalitetnim otiskom kod kuće. Nivo kvaliteta je između lasera i najboljih matričnih štampača, pre svega zbog odličnog kontrasta otiska, veoma nalik laseru. Cena je odlična za ono što nudi: da nama sada treba kvalitetan štampač za poslovnu korespondenciju, odmah bismo ga kupili umesto bilo kog matričnog štampača.

„Canon" BJ-10e zaslužuje jaku preporuku: vanserijski dobar proizvod po izvanrednoj ceni. Svaka mu čast.

COMTRAD 486-33



COMTRAD & RAČUNARI

NAGRADNA IGRA

Poklonjenom konju se u zube ne gleda, kaže stara narodna poslovica; pa ipak, teško je odoleti i ne zaviriti pod haubnu glavnoj nagradi u našoj nagradnoj igri. A da ima šta da se vidi, ima.

Na prvi pogled Comtrad 486 33 izgleda veoma impresivno. Radi se o „tower" kutiji koja, suprotno standardnoj ponudi na našem tržištu, zgleda veoma lepo. Dizajn je ukusan i jasno pokazuje da su projektanti a i prodavci i te kako imali na umu i vizuelni doživljaj mašine. Mesta za spoljne jedinice memorije su prekrivena vratancima koja pokrivaju sve uobičajene komande. Da ne biste izgubili vezu sa računarom, tri signalne diode za radno ili

naredno stanje turbo ili kompatibilni režim i aktivnost tvrdog diska prebačene su van poklopca i stavljene na levu gornju ivicu kućišta. Dakle bez obzira na to da li su vratanca otvorena ili zatvorena, uvek možete lako saznati osnovne podatke o trenutnom stanju.

POGLED IZNAD...

Pored čisto estetske funkcije, vratanca imaju i svoj praktičan aspekt kada ih zatvorite, što bi trebalo da bude uvek kada nemate nameru da koristite diskete, ujedno koliko toliko štite disketne jedinice od nepoželjnog prodora prašine. Iskustva pokazuju da se nakon par meseci rada u mehanizmima disketnih jedinica nakupi dosta nepoželjne prašine, jer se između njih uvlači vazduh od strane ventilatora na izlazu sa napajanja. Iako vratanca naravno ne mogu u celosti eliminisati ovaj problem, svakako ga smanjuju, makar toliko da je mnogo odlje imati nego nemati ih.

Kada su vratanca zatvorena, kućišta ostavlja utisak bež monolita sa veoma modernistički oblikovanim rebrima. Kao i u slučaju vratanca, i ta rebra koja progresivno rastu po visini sa leva na desno imaju i veoma konkretnu funkciju: ona su namerno postavljen otvor za propuštanje vazduha koji se prevodi preko matične ploče od izvora napajanja, odakle se izbacuje van kućišta. Dakle, radi se o planskoj putanji vazdušnog hlađenja sadržaja kućišta.

Monitor je proizvod tajvanske firme „ProVitak" model 214S, koja do sada nije bila mnogo poznata kod nas, ali se zalaganjem firme „Comtrad" i nekih drugih prodavaca po svemu sudeći snažno probija i na naše tržište. Po dijagonali ekran ima već uobičajenih 14 inča ili 36 cm, nije uobičajen raspored komandi, a ni njihov oblik. Radi se o rotacionim potenciometrima koji su montirani tako da se nalaze ispod prednje maske, što znači da su veoma lako dostupni korisniku bez ikakvog pomeranja. Ovo je mnogo bolji aranžman od nekih drugih koji se koriste, kao recimo na AOC monitorima (komande ispod klapne sa leve strane monitora, male i teško pristupačne) ili recimo nekih „Visa/TRL" monitora kod kojih se te komande nalaze delom napred, a delom na zadnjoj strani. Svako ko je imao potrebe da prelazi iz radnog režima rezolucije 800×600 u režim 1024×768 dobro zna koliko je praktično raspolagati sa svim komandama odmah tu ispred samog korisnika.

Same komande su inače sasvim uobičajene: osvetljenje, kontrast, pomer slike po horizontali, pomer slike po vertikali i širina slike po vertikali, kao naravno i taster za uključivanje ili isključivanje. Bez iznenađenja, ali na dobroj lokaciji. Opseg delovanja potenciometara je veoma zadovoljavajući. Iako ekran ne spada u klasu potpuno ravnih, ne boluje ni od nedostatka kontrasta, ni od preterane refleksije ambijenta. Najzad, po fizičkim obimima, ovaj monitor spada u manje, što znači da se za razliku od mnogih drugih ne prolaže do suprotnog zida. U našim ionako malim stanovima ni ovo nije elemenat za potcenjivanje.

Tastatura je prilično standardan i kod čest proizvod – potpisala ju je firma „Chicony". Najbolje bi se mogle opisati kao sasvim građanski mirne, bez oduševljavajućih, ali i bez razočaravajućih efekata.

COMTRAD & RAČUNARI
NAGRADNA IGRA

...I ISPOD HAUBE

Kada se računar otvori, prvi utisak je da je sve nekako pred korisnikom. To je naravno zasluga zaista velikodušno velikog kućišta, koje omogućava upotrebu i velikih matičnih ploča, sa čak sedam mesta za razne jedinice spoljne memorije.

Matična ploča koristi procesor Intel 80486DX na zdravom radnom taktu od 33 MHz. Procesor je podržan spoljnom drugostepenom keš memorijom od 256 KB u vidu brzih SRAM čipova (20 nS). Na ploči se nalaze dve memorijske banke, od kojih je jedna popunjena SIM modulima sa ukupno 4 MB sistemske memorije. Pošto ploča podržava i SIM module od 4 MB, maksimum koji se na ploču može neposredno staviti iznosi 32 MB, objektivno sasvim odvoljno za bilo koji posao koji bi se na ovoj mašini mogao raditi.

Ali to nije sve. Jedan slot se sastoji od dva funkcionalna dela. Prvi čini standardnu ISA utičnicu za proširenja, a u saradnji sa drugim delom, koji je sasvim pomeren ka dnu ploče, ovaj slot se može koristiti za posebnu memorijsku kartu sa opet 32 MB, čime bi sistem raspolagao sa impozantnih 64 MB. Malo je korisnika kojima bi trebalo ovoliko memorije, ali nije na odmet znati da može ako treba.

Veznu elektroniku obezbeđuje sada već dobro poznati i uhodani paket čipova firme OPTI, dok se o magistrali brine čip sa oznakom 82C207. Njega u potpunosti kontroliše BIOS firme AMI, koji omogućava programiranje raznih aspekata rada. Recimo, imate mogućnost da pomoću šifra zatvorite računar za sva sem vas (pazite, početna šifra je AMI), a možete programirati i brzinu radnog takta i broj stanja čekanja na magistrali. Jedna napomena – ako niste vični ovom poslu, nemojte se upuštati u izmene parametara, a ako se ipak zaglavite, ne brinite: na sigurnom ste. Ml no isključite računar, pritisnite taster „Ins" i uključite mašinu, sve će se vratiti na početne parametre i, naravno, radiće.

Tvrdi disk je proizvod čuvene firme Quantum, a model je svima nama dobro poznati LPS105AT. Quantum je inače firma koja ima najmanju stopu vraćenih diskova u industriji, manje od 0.5% (prosek je negde između 1 i 2 %). Ovaj disk ima veoma nizak profil, a druga velika vrlina mu je izuzetno mala potrošnja struje od svega 6 VA. Mala potrošnja znači i malo stvorene toplote, a to se prevodi u dugi očekivani vek trajanja, inače naveden kao 150 000 radnih časova. To je kao kada ga ne biste gasili celih 17 godina. Naravno, disk je i veoma zadovoljavajuće brz, a sudeći po kumulativnom iskustvu, i veoma tolerantan prema raznim ambijentima. Nije ni čudo što se proizvodi ovog proizvođača sa drugačijim etiketama koriste i u mašinama koje prave Hewlett Packard i Apple. Izbor „Comtrad"-a je sasvim logičan.

Sa formatiranim kapacitetom od 105 MB, disk zadovoljava veliku većinu potrebe korisnika pojedinca, bez obzira na princip da čovek nikada nema previše prostora na disku. Uz sasvim malo pažnje, 105 MB je sasvim dovoljno.

Najzad, obe flopi disk jedinice, i ona od 5.25 i ona od 3.5 inča su generički proizvodi, sasvim nalik mnogim drugima u svojoj klasi.

Ostaje još video sistem. Od obilja video kartica koje se danas nude, „Comtrad" se opredelio za bezuslovno najpopularniju kod nas, a to je ona kartica koja koristi Trident 8900 video čip. Na sreću ili po nameri, radi se o verziji kartice koja koristi odvojene oscilatore za različite funkcije, umesto generisanja taktova iz jednog jedinog oscilatora. Onako kako je isporučena u računaru, kartica je opremljena sa 512 KB video memorije, sasvim odvoljno za sve radne režime, od tekstualnog do 1024×768, u 256 boja u svim sem u najvišem režimu. Dakle, sasvim dovoljno za sve stvarne potrebe.

Monitor spada u prilično uobičajene proizvode u svojoj klasi. U saradnji sa video kartom korisnik vidi sasvim korektnu sliku, koja se nalazi u domenu očekivanog na ovom nivou danas, sa izuzetkom boje. Naime, kvalitet ili čistoća odje odskače od proseka taman toliko da to čovek primeti. Ako budući vlasnik bude radio sa odjom, imaće odsta razloga za zadovoljstvo.

PERFORMANSE

U slučaju ponašanja računara u radnim uslovima, obavili smo uobičajena merenja, ali razlozi odn tona nam ne dozvoljavaju da ih prokomentarišemo na uobičajeni način.

Ipak, konstatacija je sledeća: postoje računari koji su ispod, na ili iznad proseka. Pod prosekom se smatra nekakva nulta tačka dobijena sabiranjem rezultata merenja svih sličnih računara, koja se zatim podeli sa brojem mašina – ukratko, aritmetički prosek. Srećnog dobitnika će zanimati da zna da će raspolagati računarom koji se u apsolutno svim tačkama nalazi iznad proseka, kad manje, kad više, ali uvek – iznad.

Veoma je odbra sprega računar-disk, čak i sa sasvim standardnim podešavanjima. Uz malo mudrih intervencija, moguće je već dobre rezultate još malo poodjšati.

Propusni kapaciteti na magistrali su veoma zadovoljavajući, kao i preciznost izvora napajanja, koji ne samo što ima veoma solidne kapacitete od 220 VA, nego i sve potrebne izvode u veoma uskim tolerancijama. Vrlo, vrlo odbro.

NEKOME ĆE SE POSREĆITI

U ove teška vremena nekom srećniku će se zaista posrećiti. Jedan čitalac će odbiti računar koji spada u najvišu klasu personalaca, mašinu koja je sposobna da se hvata u koštac sa svim problemima sa jedne i mašinu koje gotovo da i nema plafon dalje nadgradnje sa druge strane.

U sadašnjem obliku to je kvalitativno natprosečna mašina, koja je pravljena više po logici kvaliteta i trajnosti nego po logici apsolutnih performansi: u prilog ovoj tezi govori niz detalja, od izbora zaista odličnog kućišta do izbora diska, pa čak i matične ploče. Nema tu igranja ili eksperimentisanje: sva su to poznati, dokazani i u praksi provereni delovi.

Budući vlasnik nije ograničen ni po mogućnostima širenja, od dodavanja memorije do ugradnje maltene bilo čega što poželi. Niti će mu biti problem prostor, niti mogućnosti pouzdanog i bezbednog rada u kontekstu napajanja strujom. Dakle, sve kockice se lepo slažu u jedan atraktivan mozaik.

To još niste videli!

Jedinstvena prilika da najzad dobijete računar a kame ste sanjali!

RAČUNARI NA POKLON 486 • 386 • 386SX

• Od prošlog broja velika nagradna igra u četiri broja Računara" • Ukupno 33 lepe nagrade, od kojih tri izuzetne vrednosti • U svakom broju kupon sa tri pitanja • Javno izvlačenje 30. septembra

U saradnji sa uglednom firmom COMTRAD naš časopis je u prošlom broju pokrenuo nagradnu igru koja traje četiri meseca i čiji će rezultati biti objavljeni u oktobarskom broju „Računara". U igri će biti dodeljene ukupno 33 nagrade, među kojima su tri vredna „Comtrad" računara: 486, 386 i 386SX.

Igra se odvija ovim redom:

- U prošlom broju (82 – maj 1992) najavili smo igru i objavili prvi nagradni kupon sa pitanjima.
- U ovom broju (83 – jul/avgust 1992) prikazujemo prvu nagradu (486) i objavljujemo drugi nagradni kupon sa pitanjima.
- U broju 84 (septembar 1992) prikazujemo drugu nagradu (386) i objavljujemo treći nagradni kupon sa pitanjima.
- U broju 85 (oktobar 1992) prikazujemo treću nagradu (386SX) i objavljujemo četvrti nagradni kupon sa pitanjima.
- U broju 86 (novembar 1992) objavljujemo rezultate javnog izvlačenja (foto-reportaža).



1. Pravo učešća i mogućnost dobijanja neke od nagrada imaju svi čitaoci „Računara". U nagradnoj igri ne mogu da učestvuju lica zaposlena u „Comtradu" i „Računarima", odnosno BIGZ-u, članovi njihovih porodica, kao ni lica koja su redovni saradnici ove dve firme.
2. Nagradna pitanja objavljuju se u četiri dela (sa četiri odgovarajuća kupona sa pitanjima) u „Računarima" broj 82 (maj 1992), broj 83 (jun 1992), broj 84 (jul/avgust 1992) i broj 85 (septembar 1992). Učesnici treba da štampanim slovima ili mašinom tačno odgovore na postavljena pitanja u sva četiri nagradna kupona.
3. Za dodelu nagrada u obzir će se uzimati jedino nagradni ankalni kuponi iz „Računara".
4. Sva četiri nagradna kupona treba – isključivo zajedno! – dostaviti na adresu: RAČUNARI, Bulevar vojvode Mišića 17, 11000 Beograd, sa obaveznom naznakom NAGRADNA IGRA. Rok za dostavu: 27. septembar 1991. (uvažava se poštanski žig)
5. Odgovori mogu da se šalju isključivo u standardnim kovertima manjeg ili srednjeg formata. U obzir za dodelu nagrada neće se uzimati veliki koverti (formata A 3 ili veći), koverti pravljeni u vlastitoj režiji, paketići i slično.
6. Redakcija „Računara" i „Comtrad" kao organizatori ne mogu da snose nikakvu odgovornost za eventualni gubitak, oštećenje ili kašnjenje vašeg pisma u pošti.
7. Izvlačenje imena dobitnika nagrada izvršiće se javno 30. septembra 1992. godine. O eventualnom minimalnom pomeranju ovog datuma učesnici će biti obavešteni tokom trajanja nagradne igre.
8. Kompletni rezultati izvlačenja će biti objavljeni u oktobarskom broju (86) „Računara".

NAGRADE

I nagrada — **COMTRAD 486**

33 MHz, 4 MB RAM, hard 105 MB, Super VGA monitor, dva flopija, tastatura, tower kućište, miš

II nagrada — **COMTRAD 386**

33 MHz, 4 MB RAM, hard 105 MB, Super VGA monitor, dva flopija, tastatura, tower kućište, miš

III nagrada — **COMTRAD 386SX**

20 MHz, 1 MB RAM, hard 40 MB, Super VGA monitor, dva flopija, tastatura, baby tower kućište, miš

Utešne nagrade

• 10 disk jedinica 3.5"/1.44 MB • 10 miševa • 10 kutija disketa

KUPON 2

Molimo da štampanim slovima upišete potrebne odgovore:

1. Računar Comtrad 486 se isporučuje sa ____________ sekundarnog keša
2. U računar Comtrad 488 ugrađen je tvrdi disk ____________ kapaciteta ____________ megabajta
3. Video kartica se zasniva na ____________ video čipu

Ime i prezime ____________

Ulica i broj ____________

Pošt. broj i mesto ____________

Kupon sačuvati i poslati na kraju igre zajedno sa preostalim kuponima na adresu: RAČUNARI, Nagradna igra, Bulevar vojvode Mišića 17, 11000 Beograd



Ne propustite sledeći broj „Računara" i svoju šansu da dobijete 486 – 386 – 386SX!

Kontakt adresa

COMTRAD
11000 Beograd, V. Popovića 6
Poslovni centar Genex
Tel. 011/2224-151, 011/2222-652
Fax 011/2223-951

LQ ZA SVAKOGA

Štampači japanske firme „Seikosha" oduvek su bili poznati kao „popularni", odnosno jeftini, ali i po sasvim prihvatljivim karakteristikama. Nova serija modela ove firme, kojoj pripada i 24-iglični model SL-92 Plus, samo potvrđuje ovaj imidž, ali i donosi prijatna iznenađenja.

Marko Kirić

Ime „Seikosha" je u našim krajevima sadašnjim i bivšim poznato još od prvih dana proboja računara u kuće, tačnije rečeno, još iz vremena kada smo za čika Klajvov „Spectrum" plaćali tek nešto manje nego što danas košta 286-ica. „Seikosha" se tada pojavila na tržištu sa svojim modelom štampača za čuvenu „dugu" kojim je potpuno potisnula originalni Sinklerov (Sinclair) model. Zbog toga ovo ime mnoge asocira na same računarske početke, a ostalima u svakom slučaju predstavlja referencu za kućni štampač prihvatljivih karakteristika i popularne cene.

Ovakav renome se, međutim, pokazao kao dvosekli mač – oni koji su mogli da biraju su zbog toga bili skloni da o ovim štampačima govore sa izvesnom dozom omalovažavanja. Vremena su se promenila, negde na bolje, negde na gore, ali stvari u svakom slučaju nisu više iste kao pre – u „svetu" su masovna elektronika i tehnika masovne proizvodnje znatno napredovali, a kod nas se ekonomska moć istopila čak i brže od teritorija. Stara slava i novi uslovi su bili dovoljni razlozi da bacimo podroban pogled na novi model „Seikosha" SL 92 Plus.

NEŽNA MEHANIKA

Opis započinjemo mehanikom, budući da je to prva stvar koja kod jednog štampača pada u oči, a u ovom slučaju bi se najkraće mogla opisati kao plastika uz nešto pomalo metala. Mada same reč „plastika" obično ima neprijatan prizvuk, u ovom slučaju za to nema nikakvog razloga – kućište je napravljeno od veoma solidne plastike i veoma lepo dizajnirano i precizno spojeno. Već na prvi pogled je jasno da je u dizajn uloženo dosta truda, a roditelja se ne bi postideli ni mnogi skuplji i izvikaniji štampači. Ivice su lepo definisane i blago zaobljene, a gornja površina je podeljena između tamnog providnog poklopca koji skriva glavu i traku za štampanje, kontrolnog panela i vodice za pojedinačne listove papira.

Ispod poklopca se nalazi glava za štampanje sa hladnjakom ne baš impresivnih dimenzija, kaseta sa trakom, a tu je i gumeni valjak. Glava za štampanje se kreće po solidnoj metalnoj osovini (koja je jedan od retkih metalnih delova u celom uređaju) pogonjena standardnim gumenim zupčastim kaišem. Mada dimenzije glave, kao i pogonskog kaiša, ne ulivaju neko preterano poverenje, tokom testa nije bilo ni jednog jedinog problema. Kaseta sa trakom je slična standardnim nepokretnim trakama kakve koristi i „Epson", mada nije sasvim ista. Ovo je u svakom slučaju pozitivna stvar, budući da u modu sve više ulaze minijaturne kasete koje se kreću zajedno sa glavom. Traka za štampanje ima sasvim konačne dimenzije, tako da postoji granica ispod koje se ne može ići bez uticaja na kvalitet. A glavna osobina svake trake, pod pretpostavkom da je izrađena od standardnog materijala, je trajnost, pošto trajnost zavisi od dužine, jasno je da je za korisnike, naročito u našim uslovima, veća kaseta u koju može da stane i duža traka daleko povoljnije alternative, bez obzira na cenu.

Prva stvar koju sam poželeo da uradim po priključivanju štampača na računar je bilo podešavanje DIP prekidača. Tu me je dočekalo prvo neprijatno iznenađenje – jednostavno ih nigde nije bilo. Posle dosta razgledanja, obrtanja i zavirivanja, dilemu je razrešilo inače prilično dobro napisano i dizajnirano (dvojezično) uputstvo: da bi se do njih došlo, trebe prvo skinuti kasetu sa trakom, pomeriti glavu za štampanje u krajnju desnu poziciju, a zatim skinuti plastični poklopac sa osiguračem na dnu štampača. Ovo se pokazalo kao veoma pipava operacija, budući da je osigurač prilično čvrst, a sam poklopac deluje prilično slabašno.

Po završenoj instalaciji celu proceduru je trebalo ponoviti u obrnutom smeru, što takođe nije prošlo bez nerviranja i prljanja prstiju, budući da se traka dosta teško namešta ispod glave. Zaista mi nije jasno zašto je pristup DIP prekidačima načinjen ovoliko nepristupačnim – prosto dobijam utisak da je konstruktor želeo da korisnika odvrati od čestog (ili čak bilo kakvog) menjanja njihove pozicije, i to tako da mu se cela operacija smuči već posle prvog podešavanja. Ovo je zapravo rešenje koje se primenjuje kod štampača sa pokretnom trakom, tako da nema potrebe za demontiranjem pola štampača, a još više čudi to što je jedino kod prihvatio „Epsonov" sistem uveden još pre dve godine, po kome su DIP-ovi smešteni ispod malog poklopca na prednjoj ploči. Ovo je detalj koji najviše smeta kod ovog modela.



Za svaki računar – za svaku kuću: Seikosha SL-92 Plus

SEIKOSHA SL-92 PLUS:

Veliki mali printer s
1 9 ugradjenih fontov
Brzina mu je prihvatl
veoma dobar i kvalite
Pogodan je pre svega

FONTOVI, OTISAK, BRZINA

„SL-92 Plus" je, u skladu sa savremenim trendom, opremljen sa devet fontova (S Roman, S H, Courier 10, Prestige Elite, Script, OCR A, OCR B, Gothic 12 i Orator) u veličinama od 10 do 20 cpi, ravno u LQ režimu, gde postoji i mogućnost proporcionalnog ispisa. Slovni otisak je iznenađujuće dobar i jasan u svim veličinama, a mogućnost dobijanja specijalnih efekata kao što su *Outline* i *Shadow* (konturna i zasenčena slova) iskorišćena je do maksimuma. Fontovi i veličine se mogu birati softverski iz programa ili direktno sa kontrolnog panela. Sa kontrolnog panela se ne mogu birati specijalni efekti, ali zato mogu sva veličina i svi fontovi. Na prvi pogled se niko ne bi rekao da ovako spartanski kontrolni panel može da pruži išta više osim prebacivanja štampača u ON/OFF LINE, izbora između LQ i draft ispisa i uvlačenja/izbacivanja papira, ali nije tako. Na veoma lepo dizajniranom poklopcu glave je naštampan uzorak koji je u prvi mah izgledao kao reklamna nalepnica koja ukratko objašnjava mogućnosti štampača, međutim, ubrzo je postalo jasno da su to zapravo oznake neke vrste menija sa glavom za štampanje kao pokazivačem. Naime, preko određene kombinacije tastera sa kontrolnog panela (ima ih ukupno četiri) se sukcesivnim pritiscima vodi glava za štampanje dok se ne zaustavi ispod oznake željenog fonta ili veličine. Vrlo originalno i do sada neviđeno rešenje.

Na brzinskom testu „92 Plus" je pokazao sasvim očekivane razlike: deklarisane brzina iznosi 200 cps u draftu i 66,7 cps u LQ modu, ali izmerena brzina iznosi 136 cps u draftu i 58 cps u LQ režimu. Iako su fabrički podaci o brzini redovno veoma optimistički nastrojeni i skoro nikad se ne poklapaju sa stvarnim vrednostima, naročito u draft režimu, postignuti rezultati pokazuju ipak preveliko odstupanje u draftu, ali vrlo malo u LQ režimu. Brzina u draftu se, s obzirom da je reč o 24-igličnom štampaču, može svrstati u donju granicu proseka, dok se brzina u LQ načinu može smatrati prilično dobrom, naročito u poređenju sa osnovnom klasom. Imajući u vidu skromnu veličinu glave za štampanje i relativno nežnu mehaniku, ne bi trebalo očekivati bog zna šta od grafičkog otiska. Međutim, „92 Plus" je i tu veoma prijatno iznenadio – krugovi zaista izgledaju kao krugovi, sa jedva primetnim izobličenjem dodnše, ali ipak mnogo manjim od očekivanog, dok su nijanse sivog zaista veoma dobro iscrpale. Na test otisku su se stopile samo dve najtamnije nijanse sivog, dok su crna, bela i sve ostale nijanse jasno razdvojena i prikazane u prilično vernom odnosu.

POVOLJAN UTISAK

Sve u svemu, osnovni utisak koji ovaj štampač ostavlja je vrlo povoljan. Na prvi pogled se vidi da nije napravljen sa bilo kakvom većom ambicijom osim da bude pristupačan i da svom korisniku omogući kvalitetan i pouzdan ispis, što je više nego ispunio, čak i prebacio. Ne verujem da bi mogao da izdrži velike količine štampe svakodnevno, ali ruku na srce, korisnici sa takvim zahtevima bi trebalo da imaju bar dvostruko dublji džep. Očigledno je da je model zamišljen kao ekonomski kućni štampač, sa svim prednostima i manama takvog pristupa. Mane su svakako nežna konstrukcija i relativno sporo štampanje, uz poseban naglasak na katastrofalni smeštaj DIP prekidača, a prednosti su niska cena (među najnižima u 24-igličnoj klasi) i male dimenzije, uz standardno dobar otisak koji bi dobro pristajao i mnogim skupljim modelima.

Ako se vaše potrebe kreću u granicama od nekoliko stranica do par desetina stranica dnevno, ako vam nije preterano bitna brzina i ako želite širok izbor fontova i pristojan grafički otisak, a ne želite ili jednostavno ne možete da izdvojite više para, onda bi ovaj štampač mogao da bude pravi izbor. Dobro ga razmotrite.

Lična karta

Dimenzije (ŠxVxD) 382 x 121 x 282 mm
Težina 3,9 kg
Broj iglica 24
Papir Maksimalni format A4 (pojedinačni listovi, beskonačni papir), 53-81 g/m²
Veznici Centronics
Emulacije Epson LQ 850
Tekstualni modovi Draft i LQ u veličinama 10, 12, 15, 15, 17 i 20 cpi
Grafika
– 8-iglična, maksimalno do 240 x 180 dpi
– 24-iglična, maksimalno do 360 x 180 dpi
Brzina
(Deklarisana/izmerena, pri 10 cpi)
– Draft. 200/136 cps
– LQ 66,7/58 cps

Korisna adresa

COMTRAD
11000 Beograd, V. Popovića 6
Poslovni centar Genex
Tel. 011/2224-151, 011/2222-652
Fax 011/2223-951

TAKO TO RADI BIL

Američki časopis „Buisiness Week" je nedavno svoju priču sa naslovne strane posvetio „Microsoftu" i Bilu Gajtsu (Bill Gates) - kompaniji vrednijoj i od čuvenog „General Motorsa" i njenom čoveku br. 1 kao zvanično najbogatijem Amerikancu današnjice, čija se „težina" procenjuje na sedam milijardi dolara.

Pet hiljada „Microsoftovih" službenika okuplja se u kaverni „Kingdom" u Sietlu. Svetla se polako gase a snop reflektora prati crvenu „korvetu" na čijim vratima je ispisana samo jedna reč: *Windows*. Iz kola izlazi „Microsoftov" potpredsednik Stiven Balmer (Steven A. Balmer), skače uvis i počinje, zajedno sa okupljenom masom da skandira: *„Windows, Windows, Windows, Windows."* Svetlost reflektora skače na „edsel" oblepljen natpisima „OS/2" (IBM-ov konkurentski paket) koji krči put u arenu. Zatim, dok iz zvučnika grmi pesma pod nazivom *Leader of the pack*, u centar pažnje grmeći ulazi deset „harli-dejvidsona" sa motociklistima obučenim u kožne jakne. Njihov vođa je suvonjavi čovek buljavih očiju, Viljem Gejts III (William H. Gates III), pozdravljen hukom mase, baš kao zvezda rok-koncerta ili fudbalskog tima na derbiju.

PRAVI VOĐA GRUPE

Zaista, pravi vođa grupe (*Leader of the pack*).

Gejts zaista ima razloga da se hvali. Tokom proteklih sedamnaest godina, od kako je osnovao „Microsoft", njegova deviza je bila: „Zapošljavam pametne ljude širokih pogleda koje izazivam da razmišljaju. Od njih zahtevam da budu spremni na ozbiljan rad".

Novajlije obično moraju da se zadovolje ograničenom platom i radnom nedeljom od 60 pa čak i 80 časova uz minimalnu obuku, koja se, po rečima Pita Higinsa, jednog od potpredsednika kompanije, obavlja uz posao. „Bacimo ih u maticu i poželimo im sreću", kaže on. Zauzvrat, zaposleni nisu sputani gomilom propisa i birokratijom. Oni koji ne uspeju da se snađu bivaju „zamoljeni" da pronađu drugi posao unutar kompanije, ili odlaze u roku od dve godine. Ovo pravilo važi bez obzira na rang – u to se lično uverio i Majkl Holmen (Michael Hallman), „Microsoftov" predsednik i bivši „Boingov" direktor računarskog odeljenja, koga je Gajts lično otpustio u martu 1990.

Oni koji uspeju da se uklope bivaju bogato nagrađeni. Tomas Dimitri, programer-pripravnik, kako bi se njegov status najbolje mogao definisati samoupravnim rečnikom, koji je odbio mnogo bolje plaćen posao da bi se tokom prošle godine zaposlio kod „Microsofta", čak je i u tako kratkom roku uspeo da napravi profit od 150 000 USD, koji će mu 1995. biti dostupan u opcijama. Pored toga je dobio i specijalni bonus za pokazane rezultate (ili „stimulaciju", kako bi se to kod nas reklo) u visini 15% svoje plate svakih šest meseci. „Microsoft" je do sada napravio milionere (dolarske, naravno) od preko 100 svojih službenika – neki od njih su jednostavno uzeli novac i otišli – Skot Oki, glavni rukovodilac marketinga, čekao je 10 godina da bi se početkom ove godine penzionisao u svojoj 43. godini sa zalihom od 40 miliona dolara.

Ali veliki broj ipak ostaje – 6% onih koji odu iz „Microsofta" je po industrijskim standardima veoma nizak udeo. Kao i njihov šef, zaposleni u „Microsoftu" više cene uspeh nego nagradu. U potrazi za pravim ljudima, poseban tim obilazi univerzitete i razmatra i po 10 000 rezimea mesečno! Ono što traže nije ni radno iskustvo ni briljantne ocene niti socijalni status, već čisti talenat. „Mi zaziremo od ljudi koji nose skupoceni nakit i elegantna odela", kaže Džon Nilson, šef „Microsoftovog" njujorškog odeljenja za prodaju. Od kandidata se pre svega traži da reši mozgalicu koja otkriva njegov način razmišljanja, umesto dokazivanja poznavanja na primer objektnog programiranja.

BILOVA INKVIZICIJA

Kao što nije svako za „Microsoft", tako očigledno ni „Microsoft" nije za svakoga. Veterani velikih korporacija često doživljavaju kulturni šok – Ričard Sigel, direktor marketinga je tokom svoje prve radne nedelje video grupu programera kako, odeveni u kupaće gaće i igrajući odbojku u hodniku, diskutuju o bagovima u nekom programu. Za osam godina provedenih u jednoj velikoj i uglednoj kompaniji nije sreo ništa ni približno nalik ovome.

"Microsoft" nije pravo mesto ni za osetljive ili plašljive. Zaposleni pričaju o „Bilovim sastancima", na način koji sugeriše da je to nešto samo malo bolje od španske inkvizicije. Gejts zasipa svoje službenike tehničkim pitanjima, izaziva, presuđuje, pronalazi zamerke, počev od pogrešnog algoritma do loše postavljenog tržišnog plana. Zbog toga nije retkost da se ljudi nedeljama pripremaju i čak drže grupne vežbe za samo jedan jednočasovni sastanak sa Gejtsom. „Ovi sastanci postižu cilj", kaže jedan od programera. „On se usredsređuje na negativne stvari i u stanju je da vam istera dušu na nos, a da na kraju kaže: 'Hej, dobar si!'"

"Ja imam najveći pojedinačni uticaj u ukupnoj korporacijskoj kulturi", priznaje sam Gejts. On ipak ne očekuje od ostalih da ga slepo slede u svojoj bezgraničnoj odanosti poslu. Njegova kola, pretrpana kesama sa brzom hranom, dolaze na parking u 9.00 ujutro i najčešće tamo ostaju do ponoći. Kad dođe kući, Gejts piše beleške još najmanje dva sata na svom kućnom kompjuteru. Okoreli neženja, Gejts ne pokazuje ni najmanje interesovanja za tako profane stvari kao što su frizure ili moda. To mu, kao tipičnom harvardskom proizvodu, omogućava potpunu intelektualnu slobodu. Kada se priprema za „Appleovu" tužbu, on postaje ekspert za intelektualnu svojinu, kada ga zainteresuje Napoleon, on ne čita jednu, već 12 biografija. Glavna preokupacija su mu ljudi i otkrića sa dramatičnim uticajem (*dramatical impact*).

Gejts izbegava godišnje odmore, ali kad ih već prihvati, onda ih pretvara u „nedelje razmišljanja". Izolovan u šumi, okružen konzervama, čita gomile knjiga i časopisa i planira „Microsoftovu" budućnost. Kompanija mu je, po svemu sudeći, iznad svega u životu – čak se i u njegovoj 10 miliona dolara vrednoj kući snova površine od preko 2 000 kvadratnih metara, koja je još u izgradnji, na sve strane nalaze natpisi „Microsoft". Kuća će, naravno, biti opremljena teniskim igralištima, ali takođe i krcata *high-tech* opremom, kako bi Gejts mogao da proverava svoje teorije. Sobe će biti ispunjene



Težak sedam milijardi dolara: Bil Gejts

redovima HDTV ekrana na kojima će se stalno vrteti digitalizovana umetnička dela, elektronske knjige i arhitektonski projekti.

SVAKO JE VAŽAN

Gde god da se nalazi, Gejts održava vezu sa svojim carstvom posredstvom elektronske pošte. Ovaj sistem koji je dostupan svim službenicima, predstavlja žilu kucavicu kompanije. Zaposleni su zapravo stalno podsticani da se uključuju i razmenjuju ideje i informacije, čak i da šalju svoje ideje direktno Gejtsu, bez nadzora svojih pretpostavljenih. Jednog ponedeljka ujutro, Nilson iz njujorškog ogranka je na svom računaru zatekao ličnu poruku od Gejtsa u kojoj mu ovaj traži informacije sa Vol Strita. „Svako može u svako doba da dobije mail od Bila G.", kaže Nilson. „Ako vas to uznemirava, onda 'Microsoft' nije za vas".

Održavanje ovakve povezanosti se sa rastom kompanije komplikuje, ali „Microsoft", najveći svetski nezavisni proizvođač softvera, uspeo je da izbegne mnoge probleme tipične za velike kompanije. Zapravo, u „Microsoftu" troše mnogo energije razmišljajući kako da kod ljudi održe osećaj da pripadaju maloj firmi. Jedan od načina je obrazovanje malih, kompaktnih grupa, što je ideja za koju Gejts priznaje da ju je „ukrao" od HP-a. Bez obzira da li je reč o programiranju ili marketingu, ni jedna grupa nije veća od 200 ljudi, a svaka se deli na podgrupe, tako da zahtevi ostanu dovoljno mali da svaki pojedinac oseća da može da utiče na konačni rezultat i da je svačiji doprinos dragocen.

Gejts veruje da je ovo kritični cilj za postizanje visoko kvalitetnog softvera. Grupa koja radi na razvoju novog NT (*New Technology*) operativnog sistema se sastoji od samo 100 programera, što zaista nije velika brojka kada

se ima u vidu da će konačni proizvod sadržati preko 2.000 000 (dva miliona) linija koda.

KAKO NAPRAVITI PONUDU?

Jedan bračni par se dvoumio oko „Microsoftove" ponude za zaposlenje – posao je izgledao privlačan, ali to bi značilo dugo putovanje, problema sa decom i gubitak udobnosti. Premišljali su se i oklevali i pronalazili gomilu razloga da to ne prihvate: nije im se sviđala 80-časovna radna nedelja, nisu želeli da putuju dalje od jedne milje (oko 1,6 km) od kuće, želeli su veliku kuću koju su mogli da plate, a uz to su i bili usred kursa karatea. Kako da to sve ostave? Jednostavno, glasio je odgovor „Microsoftovog" agenta. Rekla im je da mogu da računaju na fleksibilno radno vreme, a pronašla im je i odgovarajuću kuću na udaljenosti manjoj od pola milje od radnog mesta. Čak im je našla i novog trenera za karate. Sada oboje rade u „Microsoftu".

Čemu toliki trud? Pre svega zato što su ocenjeni kao pametni, a u „Microsoftu" pronalaženje i zapošljavanje takvih ljudi predstavlja najviši prioritet. Zbog toga „Microsoftovi" izaslanici obiđu 137 univerzitetskih naselja svake godine, neka čak i po četiri puta. Ovakva politika je značila pregledanje preko 120 000 prijava samo tokom prošle godine i direktni razgovor sa 7 400 kandidata, od čega je na kraju primljeno 2 000.

Kompanije se oslanja na stotine „tvrdih tehničara" a „Microsoftovi" programeri moraju da napišu ogroman broj linija koda – dobar programer u proseku proizvede samo 10 linija debagovanog koda dnevno, često za programe koji se sastoje i od 500 000 linija. Pored produktivnosti, najbolji programeri moraju da poseduju i veštinu rešavanja problema. Da bi pronašao ovakve talente, glavni potpredsednik Stiven Balmer razvio je posebnu tehniku koja je ušla u legendu: postavljanje neverovatnih mozgalica.

Balmerova pitanja su samo jedan od načina za ceđenje kandidata. Kandidati se dugo posmatraju pre nego što budu pozvani u centralu u Redmondu na testiranje. Kad jednom stignu tamo, provedu ceo dan u ispitivanju koja vode bar četiri čoveka iz različitih delova organizacije. Proces je toliko detaljan da je svaki ispitivač u stanju da elektronskom poštom pošalje sledećem ispitivaču sve potrebne detalje i komentare pre nego što kandidat stigne do njega.

Takav sistem je „Microsoftu" pribavio reputaciju otimača najboljih američkih mladih tehničkih, marketinških i rukovodećih talenata. Tomas Dimitri, 22-godišnjak o kome je bilo reči na početku teksta, dobijao je basnoslovne ponude od IBM-a i „Compaqa". „Već i samo 'Microsoftovo' ime sadrži neku magiju", kaže on. „Čak i samo prisustvo tog imena u vašoj biografiji izgleda fantastično".

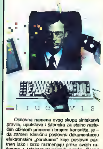
MOŽETE LI DA POLOŽITE?

Evo liste nezgodnih pitanja koja se najčešće postavljaju na razgovorima sa potencijalnim „Microsoftovcima". Od kandidata se ne traži nedvosmislen odgovor, već se tim pitanjima testira način njihovog razmišljanja.

1. Zašto su poklopci šahtova okrugli?
2. Koliko ima benzinskih pumpi u SAD?
3. Ako bi trebalo da postavite veštački pokrivač na sva igrališta (američke) „Major League", koliko kvadratnih metara materijala bi vam bilo potrebno?
4. Zašto svi automati za prodaju i džu-boksovi imaju i slove i brojke?

POSLOVANJE BEZ PAPIRA

Ujedinjene Nacije nisu samo UNPROFOR, UNICEF i UNRA, Savet bezbednosti, posmatrači i izveštači. U okviru UN rade se i druge, mnogo „pametnija" stvari, od kojih je bez sumnje jedna od najznačajnijih – UN/EDIFACT, opšte prihvaćeni standard za elektronsku razmenu podataka.



Osnovna namena ovog skupa sintaksnih pravila, uputstava i šifarnika za stalno rastućim obimom primene i brojem korisnika, je – da zameni klasičnu poslovnu dokumentaciju elektronskim „porukama" koje poslovni partneri lako i brzo razmenjuju preko svojih računara i telekomunikacionih mreža. Pored nacionalnih agencija za standardizaciju u većini zemalja Evrope i sveta, postoje specijalizovane ustanove, u celini ili delimično pod kontrolom države, koje se bave poslovima popularizacije, implementacije i stvaranje uslova za dalji razvoj ovog popularno nazvanog „bespapirnog poslovanja".

Svetski centar ovih aktivnosti je Ekonomska komisija Ujedinjenih Nacija za Evropu (UN/ECE) i njena radna grupa WP 4, sa sedištem u Ženevi, gde se u okviru posebnih grupa eksperata i u saradnji sa međunarodnim telima za standardizaciju, izrađuju nova pravila, šifarnici, uputstva itd. Eksperti za elektronsku razmenu podataka (EDI) često se sreću na mnogobrojnim stručnim skupovima širom sveta, gde razmenjuju iskustva i upoznaju se sa dometima ali i problemima u drugim zemljama. Može se sa sigurnošću tvrditi da je ovo trenutno jedna od najdinamičnijih oblasti razvoja u međunarodnim razmerama.

Koliko smo mi u sve to uključeni i kakvo je stanje kod nas? Jesmo li barem statisti u ovom filmu koji nama možda još uvek liči na naučnu fantastiku ili lude snove? Da li smo svesni svih onih fikcija koje su danas obična svakodnevnica – od Nautilusa do kosmosa, od sijalice do telefaksa – ili, naviknuti na razne utopije, čekamo da još u jednoj trci budemo prvi otpozadi?

U skladu sa svojim, na žalost, sve skromnijim finansijskim mogućnostima, Savezni zavod za standardizaciju preko svojih komisija prati ova zbivanja i preuzima standarde od međunarodne organizacije za standardizaciju (ISO). Međutim, saradnja sa UN/ECE i aktivno učešće u njenom radu nisu u nadležnosti Zavoda, već bi to trebalo da bude posao posebne agencije, koja još ne postoji, u sklopu ili pod sponzorstvom nekog ministarstva, obično za trgovinu ili ekonomske odnose sa inostranstvom, neke države na ovim prostorima koja još (ne)postoji? Nema prisustva naših eksperata u WP 4 i pored obilja nespornih argumenata od kojih je možda najuočljiviji onaj da doslovce sve zemlje bivšeg istočnog „raja" imaju svoja nacionalna tela za UN/EDIFACT, prisutne su na sastancima radnih grupa i već pokazuju prve rezultate. Od naših susednih zemalja posebno se ističe Bugarska.

I pored poznatih procena o ostvarenim neposrednim uštedama primenom elektronske razmene podataka, kao i o sekundarnim efektima kao što je povećana brzina isporuke robe i obrta kapitala, poslovanje bez zaliha, međunarodne integracija i slično, u ovom trenutku treba posebno istaći i jedan aspekt koji za zemlje u razvoju (ili u povoju) ima poseban značaj.

To je pitanje kooperativnosti, jasno izražene i operacionalizovane želje da se sarađuje na konkretnim problemima, da se prihvataju već postojeća pravila ponašanja i standardi i da se aktivno učestvuje, za svojim iskustvima, željama i potrebama, u stvaranju novih pravila i standarda. Na taj način se širi krug učesnika i dostiže se kvalitet aktivnog, kontinuiranog i od dnevne politike nezavisnog stvaranja pozitivnog imidža o svojoj zemlji i građenje novih, trajnih mostova saradnje i prijateljstva.

Početkom prošle godine Savezna vlada je pokrenula aktivnosti vezane za UN/EDIFACT, osnivanjem Komisije za koordinaciju rada saveznih organa uprave i saveznih organizacija radi olakšavanje procedura u trgovini i transportu. U izrazito nepovoljnim političkim i finansijskim uslovima, ipak se došlo do nekih početnih rezultata. Napravljeno je više prezentacija, urađena je studija „Pristup primeni EDI u Jugoslaviji", ostvarena je dobra saradnje sa mnogim velikim preduzećima, sa Privrednom komorom Beograda od. No, raspadom savezne administracije i odlaskom većine stručnjaka, postoji realna opasnost da se potpuno prekina kontinuitet i na najelementarnijem nivou, jer je čak i dokumentacija sa izveštajima o radu UN/ECE/WP 4 prestala da stiže. Sadašnje praćenje ovih aktivnosti i pokušaji da za privrednim subjektima blagovremeno sugeriše neophodnost organizacionih i tehničko-tehnoloških priprema za ravnopravno uključivanje u svetsko tržište roba i usluga, zasnovani su isključivo na entuzijazmu pojedinaca, a svode se uglavnom na rad u komisijama za standardizaciju i na poneki članak u stručnim časopisima koje niko od onih koji odlučuju i „dreše kesu" za mnogo manje važne stvari na čita.

Sve u svemu, da zaključimo u stilu neoreporterske standardne terminologije, ako to uopšte nekog zanima, kod nas je stanje aktivnosti u vezi UN/EDIFACT standarda veoma sporadično i apatično, pa ostaje da se vidi šta (ne)će biti dalje!

M. Pavlov

SVE NA JEDNOM MESTU

Ako imata dovoljno snažan hardver, ako canite svoje vreme i uz to se lako oslobađate starih navika... možda je OS/2 je prava stvar za vas. Objedinjujući na jednom mastu uglavnom sve što je u operativnim sistemima personalna informatika u stanju da ponudi, s jedne, i svojim izvornim kvalitatima, s druge strane, OS/2 vas navodi da svoj računar koristite na drukčiji način.

Zoran Životić

"OS/2 je bez sumnje odličan operativni sistem. Ono što posebno impresionira je činjenica da se pod starim DOS izgledom krije nešto potpuno drugo... S obzirom da, bar kod nas, još uvek nema pravih OS/2 aplikacija, moglo bi se reći da u ovom trenutku ipak mali broj korisnika može imati stvarne koristi od novog operativnog sistema. Tu, pre svega, mislim na programere koji dobijaju izvanredno okruženje za razvoj programa... OS/2 bi teorijski mogao da radi sa nekih 1.5 do 2 MB RAM-a, ali sa vrlo lošim efektima. Zato se za osnovnu verziju preporučuje bar 4MB, dok se u kontekstu *Presentation Manager*-a spominju i dvostruko veće vrednosti. S obzirom da trenutne cene memorijskih čipova (mada su u padu), ispada da memorije zahteva gotovo isti iznos kao i ceo računar, pa je ovo verovatno najjači argument protiv OS/2."

PRE TRI GODINE...

Verovatno ste primetili da sa prethodnim uvodom ipak nije nešto u redu. Nije u redu vremenski kontekst – radi se o izvodima iz zaključnog dela mog teksta kojim je prvi put pomenut novi operativni sistem OS/2 u „Računarima", i to tačno pre tri godine! Tadašnji OS/2 (prikazana je verzija 1.0) je pisan za mikroprocesor 80286 (mada je 386 već uveliko bio na tržištu) a mogućnosti su bile ograničene na samo jednu DOS seansu donekle problematične kompatibilnosti i, naravno, pravi OS/2 višeprogramski rad ali isključivo u tekstualnom režimu. Najpoznatije softverske kuće (Borland, Lotus, Norton) su uveliko prilagođavale svoje poznate programe za novo okruženje i pri tome izražavale puno pohvala o mogućnostima OS/2. Za tadašnje vreme enormni memorijski zahtevi se doživljavali kao ozbiljno ograničenje ali se ipak činilo da će OS/2 polako i sigurno preuzeti tržište. A onda je sve stalo.

Za ove tri računarske godine desilo nam se zaista dosta toga, ali su svakako najznačajnije pojave raznih DOS *multitasking* ekstenzija poput *Desk View*-a i, poslednjeg hita, *Windows*-a 3.0. Ovi programi su na, za većinu korisnika, pristupačniji način od tadašnjeg OS/2 nudili ono što im je sve više trebalo – rad sa više programa istovremeno i njihovu čvršću integraciju u radnu celinu. Sa ovim silnim i komplikovanim dodacima, do skoro religioznih dimenzija doveden je i pojam „potpune DOS kompatibilnost", čije je ispitivanje postalo zabava za mnoge korisnike. Konačno, doživeli smo i da konfiguracija snova od pre tri godine, 386/33MHz sa 4MB RAM-a, postane polazna platforma za svaki iole ozbiljniji rad.

Iako je OS/2 u, međuvremenu, doživeo nekoliko verzija, IBM je odlučio da tek ovu poslednju promoviše na način koji mu može obezbediti sigurniju budućnost. Dugo podgrevana neizvesnost, igranje na kartu navodnog inata IBM-a koji je odlučio da „uništi" svog dugogodišnjeg partnera Microsoft-a koji se povukao iz razvoja, anagažovanje do sada neverovatnog broja korisnika na testiranju beta verzije i puna medijska promocija konačnog proizvoda, uz vešt slogan „bolji DOS od DOS-a, bolji Windows od Windows-a" su postigli svoj efekat – o OS/2 se priča više nego o bilo čemu drugom u poslednje vreme. A šta smo dobili?

Odgovor će razočarati mnoge, ali neće biti nikakvo iznenađenje za poznavaoce IBM poslovne filozofije – dobili smo isto ono što i pre tri godine, ali prilagođeno današnjim uslovima. Tada je trebalo obezbediti kakvu-takvu DOS kompatibilnost i ponuditi novi režim koji ne mora odmah biti upotrebljen punom snagom. Danas je potrebno obezbediti potpunu DOS kompatibilnost uz DOS višeprogramski rad, *Windows* kompatibilnost i opet nešto novo, čiji značaj ne mora biti vidljiv odmah. Jednom rečju – OS/2 2.0.

NEKE VAŽNE ODLUKE

OS/2 2.0 stiže u varljivom pakovanju, koje ni iz bliza ne odaje složenost sistema koji se krije unutra – sasvim klasična kutija, na čije dimenzije smo navikli čak i kod najjednostavnijih DOS programa, uz svega nekoliko više nego tankih knjižica (najdeblja od njih, uputstvo za instalaciju ima stotinjak strana malog formata, sve zajedno nama ih „ni tri centimetra"). Jedino nas kesica sa 21 disketom formata 1.44 MB upozorava da će instalacija potrajati.

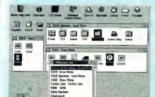
Podrazumeva se da OS/2 možete instalirati samo na nekom 386 ili jačem računaru sa najmanje 4MB RAM-a i tvrdim diskom od, recimo, bar 80-tak megabajta. Instalacija počinje ubacivanjem instalacione diskete i resetovanjem računara, kako bi se se nje pokrenuo osnovni deo samog OS/2 pod kojim se na dalje obavlja instalacija. Za mnoge vlasnike kompatibilaca ovo će predstavljati mali problem, jer moraju 3 1/2" jedinicu da postave kao disk A: kako bi *boot* sa OS/2 disketa bio moguć. Ipak, prvi ozbiljan problem će se pojaviti kada treba da odlučite koji od tajl sistema ćete koristi i kako da izaberete kombinaciju particija.

Nevolja je, naravno, u tome što pre instalacije ne znate šta možete da očekujete od novog tajl sistema – HPFS (*High Performance File System*) – jedino što iz uputstva možete saznati je da datoteke pod njim neće biti vidljive kada pokrenete DOS kompatibilni mod, što je svakako uznemiravajuća činjenica i jak razlog da odustanete od HPFS. S druge strane, karakteristike HPFS deluju veoma privlačno pa ćete, verujem, hteti da zadovoljite radoznalost i ipak ga nekako isprobate. Disk na kome sam instalirao sistem je bio podeljen na dve particije – jednu od 30MB na kojoj se nalazio DOS i potrebni alati i drugu od 90MB koju sam odlučio da preformatiram po novom sistemu.

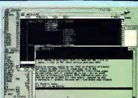
Sledeća odluka je – koju od *boot* opcija izabrati. Moguće su tri varijante. Prva je najjednostavnija – na jednu particiju se instalira OS/2 i sa te particije se sistem automatski podiže. Interesantna je takozvana *DUAL BOOT* opcija – ukoliko se vaš DOS sistem nalazi na particiji na kojoj imate bar još 30 MB slobodnog prostora možete na nju instalirati i OS/2, a zatim u toku rada, naredbama BOOT /DOS i BOOT /OS2 menjati radni operativni sistem. Ipak, u slučaju kada želite da zadržite mogućnost podizanja sisteme pod DOS-om, najprivlačnije deluje treća mogućnost – podizanje sistema preko *BOOT MANAGER*-a. Ovaj program odmah nakon uključivanja preuzima kontrolu i nudi vam izbor u obliku menija svih particija na kojima se nalazi neki operativni sistem (nije ograničen samo na DOS i OS/2 već je moguće imati istovremeno i UNIX ili neki drugi operativni sistem). Dodatna pogodnost je izbor inicijalnog sistema i vremena koje treba čekati na izbor od strane korisnika – ako ne reagujete u tom intervalu, računar se podiže pod unapred izabranim sistemom, što efektivno znači da se mašina može kompletno inicijalizovati i bez vašeg prisustva. Ova poslednja mogućnost je bila i moj prvi izbor.

Prva od opcija koju instalacije nudi je startovanje FDISK-a kako bi se izabrala jedna od pomenutih varijanti i eventualno izvršila dodatna organizacija particija. OS/2 FDISK je daleko po razumljivosti od DOS-ovog – pored standardnih opcija kao što su kreiranje i uklanjanje particija, tu su i opcije za kreiranje *boot manager*-a, dodavanje particije u njegovu listu i krajnje zbunjujuće opcije **Make startable**, **Assign C: partition** i **Set installable**. Iako nazivi deluju prilično jasno, same funkcije to ipak nisu. Moram priznati da nisam probao sve kombinacije, ali po onome što sam probao i iskustvu kolega efekti su krajnje zbunjujući – neke od ovih opcija naizgled ništa ne rade (**Assign C: partition**), a opet sa druge strane, sistem ispisuje upozorenje da pre instalacije mora postojati bar jedna particije sa atributom *Installable* od 15MB iako će se sasvim normalno instalirati i ako je particija samo *Startable*. Dodatnu konfuziju izaziva mogućnost da se *boot manager* particija (opcija, zapravo, kreira jednu malu particiju na kojoj je sam *boot* program) stavi na kraj ili na početak slobodnog prostora bez ikakvog dodatnog obrazloženja, šta se dobija sa jednim ili drugim. A nije svejedno – je sam imao sreće i pogodio opciju „na kraj", čime sam dobio normalan raspored postojećih particija – u suprotnom nastaju problemi ako ostavljate staru DOS C: particiju. Sve u svemu, instalaciona procedura u situaciji kada želite da je prilagodite postojećem stanju na disku deluje prilično konfuzno, a ni uputstvo nije baš primer efikasne pomoći. S druge strane, instalacija na potpuno „svežoj" mašini sa 500 MB diskom koji je ceo posvećen OS/2 je bila više nego jednostavna.

Dalji postupak instalacije je jasan – nakon što se prekopiraju osnovne sistemske datoteke računar se resetuje i sada se budi pod OS/2, delimično u *Presenation Manager*-u (PM), miš je aktivan, pa ostaje da klasičnim klikom izaberemo šta od celog sistema želimo da instaliramo (uz obaveštenje koliko koje od opcije zahteva prostora na disku). Jedino što možda ima smisla izbaciti radi uštede prostora na disku su igre i korisni programi koji zahtevaju oko 6MB disk prostora, ali među njima ćete verovatno pronaći i nešto što vam treba, pa je teško unapred se odlučiti da se ovo izostavi. Sledeća i zapravo poslednja od krupnih stavki je



Presentation Manager – daleko od Windows-a



Mali DOS-ovi u velikom OS/2

DOS i *Windows* emulacija koja traži oko 3 MB ali bez ovoga cela OS/2 priča gubi emisao

Nakon izbora, ostaje vam oko 20-25 minuta izmene 15 disketa i konačno izbor štampača kojima su posvećene posebna diskete Sistem sam prepozneje (mada se naravno može intervenisati u izboru) video kartu (CGA EGA, VGA, XGA i 8514) i tip miša Instalacija u ovom delu se odvija na sličan način kao kod *Windows*-a – ako imate odgovarajući drajver, možete instalirati podršku i za nestandardni hardver Sve u svemu, prvu instalaciju teško da ćete završiti za manje od sat do sat i po

PRESENATION MANAGER

Po finalnom resetovanju nakon instalacije sačekaćete više od 5 minuta da se sistem konačno startuje Ne brinite ako potraje i duže vreme je „neuhvatljivo" – čak i ponovljena instalacija na istoj mašini može drugačije da izgleda, a pogovoto na različitim mašinama OS/2 će vam ponuditi da vas provede kroz osnovni kurs rada ali ako imate imalo iskustva sa *Windows*-ima ovaj deo možete potpuno preskočiti, jer se radi o trivijalnim stvarima kao što je kontrola miša, izbor objekta i slično Konačno, ekran dobije svoj pravi izgled – *Presentation Manager* u punom sjaju!

Dizajn se prilično razlikuje u detaljima od *Windows*-a Iako važe skoro iste kombinacije komandnih tastera Iako su sve opcije iz sistemskih menija prozora tu (mada ih ima više), ipak se sistem za najansu drugačije ponaša ALT + ESC će vas kao kod *Windows*-a prebacivati iz jednog programa u drugi CTRL + ESC će dati listu aktivnih programa, ali je recimo *Windows* kombinacija ALT + TAB ograničena samo na pozore u *Presentation Manager* u – njom ne možete aktivirati DOS task koji radi u režimu „pun ekran" Možda je najveća razlika u upotrebi desnog tastera miša koji se kod *Presentation Manager*-a često koristi – uz njegovu pomoć dobijamo sistemski meni gde god da se nalazimo unutar površine nekog prozora a istovremenim pritiskom na oba dugmeta dobijamo prozor sa listom aktivnih programa

Ipak u funkcionalnom pogledu, *Presentation Manager* je nešto sasvim drugo što ćete nazreti i iz same terminologije koju IBM koristi Osnova celog sistema je objekat Sistem prepoznaje nekoliko tipova objekata – ikona, pokazivač bitmapa datoteka palete boja program itd Dominantan pojam prozora kod *Windows*-a ovde se tretira nešto drugačije U *Windows* program menadžeru otvarate prozor i u njega smeštate ikone pojedinih programa – u OS/2 kreirate objekat fascikla (*folder*) i u njega kopirate program ili bilo koji drugi objekat. Inicijalno vam OS/2 nudi nekoliko fascikli – za početak je najinteresantnija ona u kojoj se nalaze objekti „Komandni promptovi" Tu su DOS u punom ekranu DOS u prozoru OS/2 u punom ekranu i prozoru i WIN-OS/2 u punom ekranu Ostale unapred kreirane fasikle sadrže dokumentaciju korisne programe uzorke svih objekata, a tu je i nekoliko fascikli koje su inicijalno prazne – u jednu se smeštaju ikone programa koji su trenutno aktivni (umesto *Windows* rešenja u kome se ikone raporeduju u dno *Presentation Manager* prozora) a u drugu *auto-start*, sami kopirate sve ono što želite da se automatski aktivira po inicijalizaciji sistema U osnovnoj fascikli, koja zauzima celu površinu ekrana i u koju su zapravo, smeštene sve druge, moguće je takođe postaviti i odvojene objekte – tu se odmah nalazi objekat štampač disk jedinica i slično Ceo koncept će vam biti poznat ako ste imali prilika da instalirate *Norton Desktop for Windows* koji je u svemu sličan sa OS/2

KONCEPT OBJEKATA

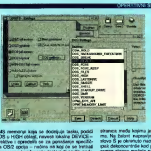
Koncept objekta je izuzetno dosledno izveden Startovanje programa je, zapravo otvaranje objekta Umesto samog objekta možete otvoriti i skup njegovih karakteristika i zatim menjati neku od njih Objekat se može obrisati kopirati ili kreirati njegova senka, koja ima identične karakteristike kao original a svaka promena bilo jednog bilo drugog se reflektuje na oba Konačno objektima se, dosledno primenjenom grafičkom interfejsu može manipulisati i pomeranjem – uhvatite objekat i prebacite ga na objekat štampač – sadržaj datoteke se štampa Objekat direktorij prebacite u neki drugi objekat direktorij i sve datoteke iz njega se na disku se premeštaju

Novi objekat kreirate bilo kopiranjem postojećeg bilo korišćenjem specijalnih objekata – uzoraka Uhvatite uzorak objekta koji vam je potreban, recimo objekat program prebacite ga u fascikla koju želite i sistem odmah otvara skup karakteristika objekta da biste podesili parametre Kod programa prvi parametar je, naravno, adresa i naziv izvršne datoteke Potom birate tip seanse (DOS OS/2 u prozoru ili punom ekranu, i WIN-OS/2) a, ukoliko se radi o DOS seansi sačekaće vas još i dodatna maska sa nizom podešavanja kojima se dotaruju i najsitniji detalji seanse, kako bi ona bila potpuno prilagođena DOS programu kome je namenjena U *General* karakteristike svakog objekta spada i ikona kojom je predstavljen – ikona se može odmah kreirati ili izmeniti, jer OS/2 poseduje mali editor za ovu namenu

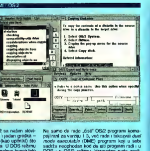
Zaista je teško predstaviti sve detalje *Presentation Manager*-a – osobina, podešavanja, načina upravljanja je zaista toliko da bi i samo nabrajanje bez ikakvih objašnjenja odnelo suviše prostora. Opšti utisak je da se radi o izuzetno razrađenom sistemu, što ću ilustrovati jednim zaista impresivnim detaljem U toku prilagođavanja *Presentation Manager* svojim potrebama, editor C EXE koji koristim sam smestio u direktorij C \EDITOR Kreirao sam odgovarajući program objekat i pri tome napisao da je izvršna datoteka C \EDITOR\C EXE U nekom trenutku sam se predomislio i odlučio da editor ipak prebacim u DOSUTIL direktorij – iz DOS-a sam uz pomoć nove interne naredbe MOVE prebacio EXE datoteku – MOVE C \EDITOR\C EXE C \DOSUTIL i zatim krenuo u *Presentation Manager* da u podešavanju objekta izmenim stazu do programa. Tu me sačekalo iznenađenje – OS/2 je „shvatio" šta se desilo i sam izmenio stazu! Naravno, ovakvo ponašanje bi se moglo očekivati da je kopiranje izvšeno iz samog *Presentation Manager*-a ali saznanje da je i DOS čvrsto povezan sa celim sistemom je ipak iznenađenje

Prethodni primer je, ujedno, i najava onoga što možemo očekivati od DOS emulacije Iako IBM tvrdi da se pokretanjem DOS PROMPT režima dobija u stvari sam DOS-a 5 0 to ipak nije do kraja tačno Ponašanje je zaista skoro identično i na prvi pogled se teško mogu uočiti neke razlika Da se ipak radi o „pisanju od početka" ukazaće i neki drugi detalji, a ne samo pomenute naredba MOVE Recimo naredba DIR na podrazumava džoker znak ukoliko se na naveda ekstenzija (pravi DOS DIR na, recimo, DIR pera daje pera txt, pera exe i sve ostale datoteke koje imaju bilo koju ekstenziju – OS/2 DOS DIR ne daje ništa) DIRCMD promenjiva kojom se specificira niz podrazumevanih prekidača naredbe DIR dozvoljava da se zada samo /O prekidač za sortiranje ispisa, bez mogućnosti da se navede dodatno slovo kojim se zapravo određuje način sortiranja Sličnih detalja ima još, ali se svakako radi o stvarima koje ni malo ne utiču na funkcionalnu identičnost – prosto ukazuju da razlika ima i da se eventualno mogu očekivati određene neusklađenosti kada sistem bude duže u upotrebi

Način na koji se DOS emulacija može podešavati je zaista vredan pažnje Sistem nudi veliki meni *DOS settings* u kome se lokalno za svaki ovakav objekat mogu podesiti i najsitniji detalji Nakraće bi se moglo reći da je tu sve ono što inače podešavate u BIOS setup-u i CONFIG.SYS datoteci u koju ste odmah na početku ubacili i neki od memorijskih menadžera Možete podesiti količinu DOS, XMS ili



Podešavanje parametara DOS seanse



Elektronska dokumentacija

EMS memorija koja se dodeljuje tasku, podići DOS u HIGH oblast, navesti lokalna DEVICE= direktive i opredeliti se za ponašanje specifičnih OS/2 opcija – načina na koji će se tretirati rezervisane kombinacije tastera ili pružiti podrška komunikacionom ili tajmer portu.

Ovde je svakako najvažnije pitanje koliko je ovaj režim stvarno kompatibilan sa običnim DOS-om. Odgovor je – u veoma visokom procentu. I zaista, problemi koji se eventualno mogu javiti pre su vezani za ipak drugačije okruženje, nego za samu emulaciju. Tako, recimo, ne možete koristi neke *Norton utility*-e čak i sa FAT sistemom, ali ne zbog toga što ne rade već zbog toga što recimo OS/2 ne dozvoljava da uklonite direktorij ukoliko je on aktivan za neki drugi task. Za mene najkorisniji NCD radi, ali će DS (*Directory Sort*) prijaviti grešku kada pokuša da zapiše rezultat sortiranja. Potreba da sistem daleko ozbiljnije vodi računa o obraćanju diskovima u uslovima višeprogramskog rada predstavlja najuočljiviju razliku, koja će vas stalno podsećati da, bez obzira što na ekranu imate samo DOS, u pozadini postoji još mnogo toga što se dešava i što vam oduzima deo suvereniteta nad sistemom.

Da je OS/2 u stanju da stvarno dobro emulira DOS govori i to da možete zaista koristiti i originalni DOS – postavite sistemsku disketu u disk jedinicu, otvorite objekat DOS FROM A i dobićete pravu DOS seansu. Opcija je prvenstveno namenjena situacijama kada treba da pokrenete neki specifični drajver, kod koga se pojavljuje problem kompatibilnosti. Umesto ubacivanja diskete, možete je „preslikati" u datoteku i zatim startovati DOS odatle – time dobijate fiktivnu disk jedinicu A koja zamenjuje onu fizičku i čiji sadržaj možete sasvim neometano menjati kao da se radi o stvarnoj disketi. Šteta je jedino što se time gubi mogućnost pristupa fizičkoj disk jedinici.

DOS se može „ugurati" i u prozor *Presentation Manager* bilo da se tako naznači u konfiguraciji objekta bilo u toku rada tasterom ALT + HOME (sivi taster HOME). Upotrebljivost ovakvog režima je velika – izborom malog fonta (na raspolaganju su veličine od 8×8 tačaka do 18×8), prozor zauzima svega četvrtinu ili petinu standardnog VGA ekrana. Mada čitkost nije velika, svakako ne dovoljna za, recimo, pisanje teksta, ipak je veoma pogodna za neke poslove – prebacivanje datoteka, startovanje programa koji ne zahteva posebnu interakciju sa korisnikom, prebacivanja podataka iz jednog prozora u drugi (radi se svim DOS programima) i slično.

Kao poslednju važnu osobinu treba napomenuti da OS/2 poznaje mehanizam kodnih stranica među kojima je i 852 sa našim slovima. Na žalost, napravljena je i jedan greška – slovo Š je okrenuto naopako (kao upitnik!) što ipak dekoncentriše kod pisanja. U DOS režimu punog ekrana možete najnormalnije koristi bilo koji od sistema koje ste i do sada koristili da dobijate naša slova na VGA karti (na Sezamu možete naći i izuzetne programe za ovu namenu koju su napisali sami korisnici Sezama), ali ostaje otvoreno pitanje naših slova u režimu rada u prozoru.

BOLJI WINDOWS OD WINDOWSA?

WIN-OS/2 je režim koji u potpunosti emulira *Windows* 3.0 i, za mene bar, je najmanje uzbudljiv element novog operativnog sistema. Činjenica je da *Windows* mnogi koriste skoro isključivo zbog DOS multitaskinga i da se pojavom OS/2 ovaj razlog gubi, svodi upotrebnu vrednost ove opcije na nekoliko standardnih aplikacija koje su u upotrebi (*Word* ili *WordpeRfect for Win* i *Corel Draw* uglavnom iscrpljuju listu).

Iako se samo startovanje odvija interno preko DOS režima (pokrene se DOS, pa onda automatski izvede winos2.com) ipak se ni ovde ne radi o potpuno identičnoj stvari već o emulaciji – pokušaj da instalirate i startuje običan *Windows* propada. Sitnih razlika koje sam pomenuo kod DOS-a ovde u odnosu na standardan *Windows*, nisam primetio. Sistem je zapravo osiromašen za stvari koje su pod kontrolom samog OS/2, pa tako nema DOS PROMPT-a e i *setup* se sveo samo na opciju lokalne mreže. Sve ostalo izleda i radi potpuno isto.

Za WIN-OS/2 režimom sam imao problem sa mišem koji nisam uspeo da rešim bez naknadne instalacije – u nekom trenutku menjajući port na koji je miš priključen i projavljujući ovu izmenu u *Presentation Manager*-u, ostao sam potpuno bez miša u *Windows* režimu. Stvar je prilično neprijatna, jer se po povratku u *Presentation Manager* miš gubi i u njemu, pa ostaje jedino da se resetom operativnog sistema miš povrati.

OS/2 NOVITETI

Na žalost, informacije o pravom OS/2 komandnom režimu su ravne nuli. Trebalo bi da se radi o pravom 32-bitnom režimu sa linearnim adresiranjem memorije, ali nama načina da se za sada proveri o čemu se stvarno radi. IBM nudi svoj C komapajler za ovaj režim, ali nam još nije raspolaganju.

Ono što sam probao je deo programa koji je bio razvijen za ranije verzije OS/2 (1.0, 1.2 i 1.3) i uvrdio da postoji potpuna kompatibilnost. Ne samo da rade „čisti" OS/2 programi kompajlirani za verziju 1.3, već rade i takozvai *dual mode executable* (DME) programi koji u sebi sadrže neophodan kod da isti program radi i u DOS i u OS/2 režimu. Verovatno niste znali, ali skoro sigurno imate i jedan takav OS/2 program. *Microsoft Word* je pisan u DME formatu, pa ga možete potpuno ravnopravno pokretati i u DOS i u OS/2 režimu.

OS/2 komandni režim zadržao je raniju filozofiju – spolja isto kao DOS, ali iznutra potpuno drugačije. Naredbe u ovom režimu već znate – od DIR, preko COMP, XCOPY do DISKCOPY. Da nije sve isto kao u DOS uverićete se ako pokrenete ovu poslednju – čak i 1.2MB disk će biti kopiran u samo jednom prolazu. Naravno, u OS/2 komandnom režimu ne može da radi NCD i mnogo sličnih programa bez kojih nam je život postao nezamisliv, pa ostaje da njegovu pravu vrednost sagledamo tek kada dobijemo alat za razvoj programa. S obzirom da je opšti trend ka grafičkom interfejsu, pitanje je koliko ćemo stvarno videti komercijalnih programa namenjenih ovom režimu – verovatnije je da će orijentacija ipak biti na *Presentation Manager*.

NOVI FAJL SISTEM

Novi fajl sistem zaista po svojim karakteristikama deluje izvanredno. Umesto beskonačnog sekvencijalnog pretraživanja direktorija u dubinu kako bi se pronašla datoteka, HPFS koristi princip indeksiranja poznat iz klasičnih baza podataka. Posledica je daleko brže otvaranje i zatvaranje datoteka, što se i te kako oseti u radu. Druga bitna karakteristika je ugrađena optimizacija na raspoređivanje klastera koji se drže koliko je to god moguće u kontinulanom nizu, čime otpada potrebe za povremenim komprimovanjem diska. I iskorišćenost prostore je daleko bolja nego kod FAT sistema, jer jedan klaster čini jedan sektor (512 bajtova). Konačno, sistem dozvoljava da datotekama dajete nazive proizvoljnih dužina (do 254 znaka), a pri tome možete koristi i velika/mala slova (samo estetski efekat, datoteku možete adresirati nezavisno od kombinacije koju upotrebite) kao i razmake između reči. Značajne novost na koju ćemo se sigurno jednog dana navići je i znatno više informacija o samoj datoteci – pored niza dodatnih atributa, programu su na raspolaganju i informacija o trenutku kada je datoteka kreirana, kada je poslednji put menjanja, pa čak i kada je poslednji put čitan njen sadržaj. Interno, HPFS poseduje još mnoge druge osobine koje tek treba da iskoriste buduću disk *utility* programi.

Još jedna od lepih osobina HPFS je što je

ipak (suprotno onome što se tvrdi u instalacionoj dokunetaciji) čitljiv i iz DOS sanase. Jedini uslov je da datotekama dajete nazive koji se mogu pročitati i u DOS, dakle bez razmaka i ograničeno na 12 znakova. Datoteka koja ne poštuju DOS konvenciju naziva jednostavno neće biti vidljiva. Čak i ako računar podignete sa DOS diskete HPFS particije će biti čitka ukoliko u CONFIG.SYS ubacite liniju DEVICE=FSFILTER.SYS i prekopirete ovaj drejver iz OS2 direktorija.

Sve u svemu, HPFS deluje kako moderen fajl sistem posebno optimizovan za rad sa velikim diskovima (IBM preporučuja da se instelire na diskova sa više od 60MB), odličnih performansi što se sve zajedno u praksi zaista i pokazuje. Kombinovan sa ugrađenim cache-om (koji koristi i keširanje pisanja) na drugoj test mašine (Šutlić 486/50MHz, 8MB RAM-a i 500MB SCSI disk) koja je konstila isključivo ovaj fajl sistem ja pokazivao značajne dobitke u brzini rada aplikacija koje često otvaraju datoteka. Konsnici Sazama i SOR neće prepoznati brzinu kojom poruka ulaze u SOR! Ipak, i pored svih lepota novog sistama posla višednevnog testiranja bio sam prisiljen da ga skionim sa svog diska i vratim se na stari FAT sistem. Da nije bilo potrebe da pišem ovaj prikaz, učinio bih to i ranije – problemi koja mi je zadavao bili su više nego ozbiljni.

Glavni problem sa HPFS-om je njegova osetljivost – ugasite li slučajno mašinu bez propisne procedura (nestanak struje na primer) dobićete u najmanju ruku haos pred kojim ste zbog napostojanja odgovarajućih alata (kraljevstvo za OS/2 Norton DISKEDIT!), potpuno nemoćni. Jedno DIR C: i dobijate naizgled bezazlanu poruku o tome kako neki interni podatak nema odgovarajuću vrednost i to je sve – ceo disk C: je nedostupan. Da stvar bude gore, ako je C jedina particija koju imate, pokušaj da pokrenete OS/2 CHKDSK će vas pozdraviti porukom kako je disk u upotebi i u tim uslovima ne mogu biti ispraljena greška! Na sreću, IBM je predvideo ovakvu mogućnost, pa možete konstiti proceduru ponovne instalacije uz odustajanje u određenom trenutku tasiarom ESC čime dobijate OS/2 prompt sa disketa i mogućnost da pokrenete CHKDSK. Ovaj program će grešku ispraviti, ali vas može, kao i mene, već pri prvom sledećem startovenju sistama ponovo sačekati iste greška prvi put kada pokušate nešto da zapišete na disk. Da ne spominjem enormno vreme potrebno za podizanje sistema – može se desiti da sačekate i više od 10-15 minuta da se pojavi *Presentation Manager*. Razlog je direktiva **autocheck** koja je ubačena u config.sys tako da sistem automatski startuje CHKDSK svaki put kada se inicijalizuje a njegov rad može stvarno da potraje bez ikakvih obaveštenja sem intenzivnog rada sa diskom.

Možda situacija na koju sam ja nalateo nije tipična, ali ipak stoji činjenica da je sasvim moguća, a jedino rešenje je tada kompletno preformatiranje particija. Konačnu odluku da skinem HPFS sam doneo kada sam iz samog *Presentation Manager* pokrenuo CHKDSK. Svaki put kada ga pokrenete dobićete obaveštenje da sa alokacijom datoteka nešto nije u redu! Nakon početne panike i nekoliko „nasedanja" na ovaj vic i ponavljanja mukotrpne procedure pokretenja OS/2 sa diskete shvetite da je razlog u SWAP datoteci koju OS/2 intenzivno koristi i za koju CHKDSK uvek prijavljuje da nema odgovarajuću informaciju o alokaciji prostora.

Sve u svemu ostaje nam da se nadamo da će HPFS ipak u sledećim verzijama postati pouzdaniji što svakako po svojim karakteristikama zaslužuje.

SAMO UZ POMOĆ HELPA

Na proteklom sajmu tehnika na koma je IBM promovisao novi operativnisistem kod nas, demonstrator je sa zadovoljstom konstatovao kako se uz program isporučuje samo minimalna dokumentacija i kako je sva ostalo ugrađeno u razrađeni help mahanizam, koji je u toku rada uvek na dohvat ruke. I zaista, HELP je doveden skoro do savršenstva sa mnoštvom unakrsnih referisanja, odvojenim sadržajem i mogućnostima najrazličitijih pretraživanja. Ipak, pokazalo se da ovakav način ima svojih mana – pre svega u toku upoznavanja sa sistemom. Lepo je što se sve može pronaći u elektronskoj dokumentaciji, ali je pravo pitanje – šta da se traži? Sa iskustvom iz ranije verzije OS/2 lako sam kroz help pronalazio tačke koje su me zanimale ali i sada nisam siguran da li se u help-u krije i neka bitno nova informacija! Ukoliko je ovaj potez IBM-a način da se smanji osnovna cena programa i ukoliko se štampana dokumentacija može nabaviti odvojeno, onda je ovakav potez zaista za svaku pohvalu. Ukoliko je pak globalna orijntacija da se izbegne papir onda bi trebalo dodati i deo koji bi detaljnije upoznao korisnika sa sistemom od postojećeg *tutorial*-a.

DOS I POSLEDNJI DANI

Da biste konačno donali sud o toma isplati li se preći na novi operativni sistem ili ne potrebno je da poznajete još nekoliko činjenica. Najvažnija među njima je da OS/2 radi kako treba tek sa 6MB RAM memonje što praktično na mnogim mašinama znači 8MB. Sa samo 4MB sistem je prinuđen da intenzivno koristi disk a tada je rad, posebno prebacivanje iz jednog programa u drugi neprijatno sporo – potrebno je ponekad i preko 10-15 sekundi za ovu operaciju. Testirajući OS/2 na 386/33MHz i 486/50MHz platformama, primetno je da brzina računara nije toliko kritična kao količina memonje. Na EISA 486/50 računaru se 8MB RAM-a OS/2 stvamo „leti", ali je rad i na slabijoj mašini više nego konforan ako imate 8MB.

Drugo pitanje na koje treba da odgovorite je da li ste u ime dodatnog komfora spremni da se odreknete nekih navika. To se pre svega odnosi na lepu osobinu DOS-a da možete da mu radite „šta vam padne na pamet" – resetujete računar kad hoćete, podižete ga sa diskete ili sa tvrdog diska, jednostavno menjate verziju DOS-a i slično. OS/2 ne spada u glomazne operativna sisteme, ali je daleko od jednostavnosti DOS-a i nameće neka pravila igre koja morate poštovati.

Ako je odgovor na oba ova pitanja – da, verujem da ćete pronaći puno zadovoljstva u radu pod OS/2. Na čuveno „crveno dugme" možete potpuno zaboraviti, jer je OS/2 u stanju da „ukroti" i najnezgodnije krahove DOS programa bez ikakvih posledica. Broj DOS taskova koje možete aktivirati je praktično neograničen (4, 5 ili 6 ne predstavljaju nikakav problem) a konkurentan rad umerenog broja programa se malo oseća iako, naravno, svaki od njih radi adekvatno sporije. Ovo se jedino ne odnosi na komunikaciju – intenzivne upotreba COM porta u nekom tasku (pre svega kod intenzivnog prijema podataka) zna znatno da ukoči ceo sistem.

Naravno, pitanje je da li vam OS/2 uopšte i treba. U razgovoru sa kolegama često su mi postavljali baš to pitanje – „a šta će ti više DOS programa istovremeno?" I zaista bez multitaskinga pod DOS-om se može, ali i ne mora! Jednom kada se naviknete na to da nije potrebno da prekinete intenzivan rad na, recimo razvoju programa da izađete iz editora i potpuno prekinete tok razmišljanja samo zato da biste formatirali disketu na koju želite da snimite zaštitnu kopiju, niti da je potreno da čekate da kompajler završi posao da biste se vratili u editor i nastavili da razmišljate o daljem razvoju programa, niti da je potrebno da sedite i čekate da program prenese podatke modemom posle sticanja takvih navika zaista se više ne postavlja pitanje „čemu multitasking".

Na kraju mi ostaje da donekle ponovim zaključak koji sam izneo i pre tri godine. OS/2 je zaista odličan operativni sistem, ali zahteva određene računarske resurse bez kojih je jednostavno neprimenjiv. Za određeni broj korisnika njegova uslovna glomaznost u odnosu na ono što im od takvog operativnog sistema treba će predstavljati odlučujući faktor da se ne opredele za prelazak na novi sistem. Možda će ponekog odbiti i sitniji bagovi kojih u sistemu ima, ali ipak ne utiču na ozbiljniju primenu. Verujem, ipak, da će OS/2 naići na odobravanja kod onih kojima je računar sredstvo rada i kojima je produktivnost faktor o kome vode računa. Ne treba zaboraviti da OS/2 zapravo nudi sve što je danas dominantno, i to na jednom mestu – i DOS, i WINDOWS i OS/2 specifični režim što i samu investiciju čini veoma isplativom.

Pitanje je koliko će IBM ovoga puta uložiti u dalja unapređenja sistema i podršku proizvodu – ako to bude činio na svoj poznati način, verujem da će OS/2 postati ozbiljna alternativa koja će nam konačno omogućiti da bezbolno ostavimo DOS iza nas.

MULTIPROGRAMSKI TEST

Da biste stekli sliku o tome kako izgleda rad više DOS aplikacija istovremeno, napravio sam jedan jednostavan test koji nije mnogo precizan ali je dovoljno ilustrativan. Kao test program korišćen je RI.BAT (Računari indeks) koji testira video podsistem, u tekstualnom režimu mikroprocesor i disk podsistem. U odnosu na standardnu varijantu test je izmenjen tako što je na početak ubačena sledeća linija:

```
:wait
if not exist \start goto wait
```

Na ovaj način je obezbeđeno da se svi programi startuju istovremeno u trenutku kada se preko jedne kontrolne seanse kreira datoteka start. Rezultati su dati u sledećoj tabeli:

	FILE	CPU	VIDEO
1	25.05	4.34	17.30
1+	25.82	4.45	10.82
2+	76.36	9.04	21.09
3+	112.42	13.30	31.71

Uz broj 1 su dati rezultati kada se test izvodi u samo jednoj i jedinoj aktivnoj DOS seansi. 1+ je test aktivran u pozadini preko kontrolne seanse (dakle dve DOS senase od kojih kontrolna služi samo da aktivira test, ona je u prednjem planu ali se u njoj ništa ne radi). Test 2+ daje prosečne vrednosti za dva DOS taska u pozadini (plus kontrolni task) i poslednja 3+ daje rezultate tri aktivne DOS seanse čija mogućnost je bila i moj prvi izbor.

Ono što prvo pada u oči je da DOS seansa u kojoj se ništa ne radi skoro ni malo ne oduzima vreme ostatku sistema. Iz ostalih testova se da zaključiti da OS/2 zahteva minimalno vreme (overhead) za prebacivanje sa jednog taska u drugi, jer vreme sa svakim novim taskom skoro linearno raste. To, na žalost, ne važi u slučaju pristupa disku koje očigledno zahteva prilično arbitriranja sistema.

Korisna adresa

PP Izvor
11000 N Beograd, Drugi bulevar 113
Tel. (011) 136-754, 135-323, 134-245

KAKO SAM OSVAJAO OS/2

Da li, u ovom teškom vremenu, pomišljate da je poslednji trenutak da promenite nešto bitno u svom životu? Za početak, pokušajte sa operativnim sistemom pod kojim radite.

Dejan Ristanović



Svi mi koji pišemo o kompjuterima, iskreno rečeno, nemamo baš previše razloga da se žalimo na DOS – da je dotični napravljen kako treba, kako bismo iz godina u godinu prelistavali i isprobavali razne „revolucionarne" dodatke i proširenja, utvrđujući koji se od njih slaže sa nekim drugim a koji, ako ga krivo pogleda u nekoj noći punog meseca, sruši sistem? Sa druge strane, mi koji programiramo PC računare već tradicionalno kukamo na DOS – manje-više i za zaštitu podataka od nekontrolisanih programa, i za multiprogramski / multikorisnički rad i za sve slične stvari, ali kada imate računar sa 8 ili 16 megabajta memorije, a u svom programu koristite svega 600 kilobajta razdeljeno na komadiće od po 64 K, tu nešto žestoko nije u redu! Istina, postoje raznorazni DOS *Extender*-i (npr. *High C*), ali na tom segmentu tržišta nema baš mnogo konkurencije, pa je i komfor prilično slab – kada se jednom naviknete na DOS kompajlere, povratak u neko „kameno doba" vam baš i neće prijati. Najzad, mi, koji koristimo računare smo negde u „zlatnoj sredini" – pričaju nam da će neki od novih operativnih sistema ponuditi brži i komforniji rad, ali i sa DOS-om se lepo živi, naročito kada se uzme u obzir njegova „uigranost" – kada vam je zadnji put DOS ispisao „Internal error 123" i zakočio sistem? Meni nije već 6 godina.

NI ON NIJE OD JUČE

OS/2, naravno, nije nastao juče – još kada je 1987. godine promovisao PS/2 seriju, IBM je najavio i OS/2 koji je zaista i izašao uz zakašnjenje od „samo" desetak meseci. No, bio je namenjen AT računarima i, kao takav, stao negde na pola puta između DOS-a i nečega stvarno novog: nikada nisam bio u iskušenju da ga ozbiljnije isprobam. Dolazile su i prolazile verzije 1.1, 1.2, 1.3 ali ni jedna od njih nije bila „prava stvar", stalno je negde u vazduhu „lebdela" priča o OS/2 2.0, koji će predstavljati prelazak sa DOS-a na „nešto drugo". Ko čeka taj dočeka, pa sam tako i ja u poslednjim danima beogradskog Sajma tehnike dočekao svoj primerak OS/2.

Sam izgled paketa me je na malo iznenadio – očekivao sam nekoliko disketa i još više knjiga, a dobio **mnogo** disketa i ni malo knjiga – nekoliko brošurica na po dvadesetak strana teško da računam u ozbiljnu dokumentaciju! Sve je, navodno, u samom OS/2, tj. u njegovim HELP-ovima što bi u piratska vremena svakako izazvalo oduševljenje, danas, takav pristup samo produžava vreme upoznavanja sa novim operativnim sistemom, jer se lutanjem po HELP-ovima lako propuštaju važne osobine programa, a ne postoji ni način da čovek na prvi pogled (prema debljini poglavlja koje je tome posvećeno) shvati koliko je neki segment važan ili obiman. No, šta je, tu je – trebalo je sesti za računar i videti šta se sve nalazi na toj dvadeset jednoj disketi od 1.44 megabajta zvanoj OS/2.

PETLJANJE S DISKOVIMA

Za test sam odvojio računar preduzeća „Šutić" zasnovan na procesoru 80486 koji radi na 50 MHz i sa svojih 8 megabajta memorije komunicira posredstvom keša od 64 kilobajta. SCSI hard disk od 210 megabajta opremljen je sa još 4 megabajta keš memorije, a za video je zadužena *Carrera* Super VGA kartica. Ukratko, „puna" konfiguracija koja, verovali ili ne, i nije daleka od onoga što OS/2 zahteva – novi operativni sistem, istini za volju, radi i na 80386 računarima, ali je 8 megabajta memorije realni minimum – sa manje od 4 M ne može se instalirati, a sa 4 M radi ali uz dosta „štucanja" i znatno smanjenje komfora. Što se, najzad, diska tiče, potpuno instalirani OS/2 zauzima tridesetak megabajta prostora, a kažu da se uklanjanjem HELP-ova i nekih drugih manje potrebnih stvari može svesti na 15-16 megabajta. Obzirom da na disku, osim operativnog sistema, treba imati i mnoge druge stvari, instalacija na diskovima manjim od stotinak megabajta teško da ima smisla. Ukratko, OS/2 kao program možda i ne košta mnogo, ali hardver potreban za rad pod njim predstavlja veoma ozbiljnu investiciju.

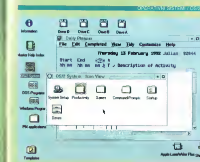
Pomenuti disk od 210 megabajta bio je podeljen na dve particije od po 100 megabajta – planirao sam da prvu posvetim DOS-u (instalirao sam na nju par potrebnih stvari, npr. *WordPerfect*, *Venturu*, *Harward Graphics*...), a na drugu da stavim OS/2 koji bi, dakako, mogao da pristupa i DOS-ovoj particiji. Kao što će se videti, ovaj plan je ostao u domenu lepih želja.

Prva prepreka koju je trebalo savladati je što u računaru nisam imao drajv od 3.5 inča ni RS-232 karticu pomoću koje bih priključio miša. Karticu sam izvadio iz nekih „unutrašnjih rezervi" (odbacio sam je pre godinu dana pošto je izgledalo da ne radi ili barem da ne umem da podesim mikroprekidače, ispalo je da mi je nedostajala dovoljna motivacija, pošto su mikroprekidači došli na mesto posle „samo" 12 pokušaja), a disk sam mogao jedino da izvadim iz nekog drugog računara. U želji da umanjim broj šrafova koje treba odvrnuti, postavio sam dva računara jedan pored drugog, otkačio kabl koji vodi ka drajvu od 5.25 inča sa jednog i prikačio ga na 3.5 inča drajv drugog. Zbog napajanja drajva trebalo je uključiti oba kompjutera (još jedna razvodna kutija...) ali je sve to skupa bilo posao od petnaestak minuta – mala cena za progres. Začudo, na instalacionoj disketi nije bilo programa INSTALL – posle kraćeg razmišljanja, zaključio sam da je jedini izlaz u tome da se ona umetne u drajv i računar resetuje, u nadi da će instalacija sama započeti. Tako je nekako i bilo, što će posebno oduševiti sve vas koji ste mislili da ćete se spasti hardverskih egzibicija – da biste instalirali OS/2 sa 3.5 inča disketa, drajv od 3.5 inča se **mora** odazivati na ime A, ako nije tako, moraćete da razmenite kablove! Bilo je nekih pokušaja da se ova operacija izbegne (korisnici Sezama su isprobali nekoliko *boot_from_B* programa), ali ni jedan od njih nije krunisan čak ni polovičnim uspehom – možda se računar može „prevariti" da startuje neku igru sa B, ali kompletan operativni sistem... malo teže. Ukratko, kada kupujete OS/2, potrudite se da ga dobijete na 5.25 disketama ili – spremite šrafciger.

KAD SE DESI ONA STVAR

Posle umetanja druge diskete i par iscrtanih (prilično efektnih) uvodnih ekrana, desila se „ona stvar" – računar pređe u tekst mod, kaže „Interna greška – TRAP 07C", ispiše sadržaje registara (svaka čast tom 80486, koliko registara ima i kolike brojčuge drži u njima – jedva sve stane na ekran!) i predlog da se javim dileru koji mi je prodao OS/2. Divota. Probao sam još dva puta, i rezultat je bio identičan. I, šta sada?

Iskreno rečeno, u nekim normalnim uslovima bih verovatno odustao i ostavio probu za neku bolju priliku, možda kroz koji mesec ili godinu, tako sam, uostalom, „preskočio" i DOS 4.00. Međutim, nikako nisam mogao da zaboravim da sam zarad ove probe rasklopio dva kompjutera, a da i ne pominjem onih 12 proba RS-232 kartice i traženje razvodne kutije, „predaja" posle takvih „žrtava" apsolutno ne dolazi u obzir. No, šta zapravo i mogu da uradim? OS/2 me pre zloslutne poruke ama baš ništa nije pitao, što znači da sam teško mogao zadati neki pogrešan parametar. Sve ovo se dešavalo u danima pre nove... blokade, dakle dolazilo je u obzir zvanje IBM-ovog tehničkog servisa, ali oni teško da bi mi nešto mnogo pametno rekli – kada god sam pokušao da se raspitam o specifičnim problemima pri instalaciji nekog softvera, pitali su me da li sam uključio računar, pa da li je priključen monitor, da li sam dobro umetnuo disketu, da li sam siguran da sam otkucao A:INSTALL a ne A:INSTALL i tako to, najzad bi me uputili na onoga ko mi je prodao hardver. Pitati na BIX-u pruža više nada na uspeh, srećom, modem je priključen na treći računar koji nisam morao da rasklapam (uvek sam govorio da čovek ne može imati previše računara u kući). No, baš dok sam se spremao da zovem JUPAK, pada mi na pamet spasonosna ideja – kada nešto ne radi pod DOS-om, prvo treba isključiti sve drajvere i rezidentne programe iz CONFIG.SYS-a i AUTOEXEC.BAT-a. Ovde nema ni jedne od tih datoteka, ali su ipak mnoge stvari „duplirane" – procesorski keš, keširanje video memorije, diska... vredi probati bez svega toga. Ušao sam, dakle, u SETUP i rešavao probleme „tehnoloških viškova": eksterni procesorski keš? Isključiti. Interni? Isključiti. Video memorija i



BIOS? Ne kešira se. Magistrala? Najsponija moguća. 4 M keša na SCSI kontroleru? Bez njih. Računar je posle svih ovih amputacija bio vidno sporiji ali je, na moje veliko iznenađenje, instalacija OS/2 prošla. Docnije sam, eksperimentišući, ustanovio da se procesorski keš može uključiti, BIOS normalno keširati, a magistrala ubrzati; jedino je SCSI kontroler morao da se odrekne svog keša. Ukratko, svaka čast IBM-u, ali OS/2 2.0 još nije prilagođen iole specifičnijim PC konfiguracijama!

TRI KLJUČNE GREŠKE

Sledeća važna odluka je *filing system* – OS/2 startuje svoj FDISK i zahteva da pripremite prostor za njegovu instalaciju. Ugasio sam, tako, *extended DOS* particiju koja je sadržala drajv D i tu postavio jednu OS/2 *primary* particiju kao i jednu *Boot Manager* particiju od megabajta. Ideja je u tome da se po svakom uključenju računara startuje *Boot Manager* koji će pitati za ime operativnog sistema koji se startuje – DOS ili OS/2; stvar se može rešiti i na par drugih načina, ali je ovaj ubedljivo najpogodniji. Osim toga, treba izabrati standardni DOS-ov mehanizam pristupa datotekama preko FAT-ova ili novi HPFS (*High Performance Filing System*). FAT mehanizam, iako jednostavan za realizaciju i razumevanje, ima nekoliko vrlo ozbiljnih mana. Pre svega, ako otvarate datoteku u nekom direktorijumu treće ili četvrte generacije, DOS mora da počne od *root*-a, pronađe prvi direktorijum, onda šeta glavu diska do njega, traži njegov poddirektorijum i tako sve dok ne nađe datoteku ili potvrdi da ona ne postoji. DOS (počevši od verzije 3.30) pokušava da reši taj problem programom FASTOPEN, *PC Tools* i *Norton Utilities* omogućavaju fizičko postavljenje direktorijuma na sam početak diska i tako dalje, ali je sve to tek blede zamena za „pravu stvar" koja se zove *hash* funkcije, odnosno BTRIEVE mehanizam implementiran u okviru HPFS-a – operativni sistem će na osnovu samog imena datoteke biti u mogućnosti da je praktično trenutno „pronađe", uz uzgredno ukidanje potrebe za kompresijom diska i sličnim operacijama. Omogućeno je, osim toga, davanje dugih imena datotekama (u imenu se mogu pojavljivati čak i blanko simboli, ali se tada ime mora kucati pod navodnicima, npr. DIR „OS2 USLUŽNI PROGRAMI") i uz svaku datoteku se ne pamti samo jedan datum već njih nekoliko – odnose se na kreiranje, poslednje modifikovanje i druge zanimljive podatke iz „lične karte" datoteka. Sve to lepo zvuči, ali ime i svoju cenu – iz „pravog DOS-a" ne možete pristupati ovako formiranoj particiji (iz DOS prozora samog OS/2, naravno, možete), osim uz pomoć specijalnog SYS drajvera koji meni, uzgred budi rečeno, nikako nije proradio. No, kao je bal neka je i zabave – izabrao sam HPFS, sačekao da se disk formatira i uskoro dobio lepu HPFS OS/2 particiju od 100 megabajta.

Pokazalo se, na žalost, da sam u ovoj fazi reda napravio tri greške, zapravo, svaka donete odluka se pokazala pogrešnom! Pre svega, **nemojte** se opredeliti za HPFS particiju. Svi pomenuti razlozi za i protiv stoje, ali se pojavljuje još jedan koji začas „pretegne" na FAT stranu: ako nešto pođe naopako, tj. ako se OS/2 „zaglupi" (što se dakako zna desiti!), nemate alatki pomoću kojih biste oporavili oštećenu particiju i tako eventualno spasili svoje podatke! Ukoliko se, međutim, opredelite za FAT particiju, tu su *Nortonovi*, *Central Point*, *Mace* i ostali „spasioci" na čelu sa čuvenim *Norton Disk Doctor*-om – izgledi za oporavak podetaka su jako visoki. Izbor HPFS-a je, dakle, bila prva greške. Druga je što sam *Boot Manager* stavio ispred, a ne iza OS/2 particije – iskreno rečeno, ne znam zbog čega je ovo greška, ali svi na Sezamu i mnogi na BIX-u kažu da je treba staviti iza OS/2, tj. na sam kraj diske. Niko nema neki dobar razlog za to, ali kad svi kažu... Treća i najveća glupost je bila instalacija **primarne** OS/2 particije – obavezno se opredelite za **extended** particiju. U protivnom, dolazite u situaciju da se OS/2 disk zove C baš kao i DOS disk ako butujete sistem pod DOS-om. Ukratko, iz DOS-a ne možete pristupiti OS/2 particiji (što je i očekivano, ako se izabere HPFS), ali ni iz OS/2 ne možete pristupiti podacima na DOS particiji C. Da bi stvar bila još lepša, izgleda da se *primary* particija iz FDISK-a ne može pretvoriti u *extended* uz čuvanje podataka na njoj, ukratko, DOS i OS/2 su na mom računaru bili dva sveta, što za probanje ne predstavlja posebnu smetnju ali za stvarni rad nije ni malo prijatno.

LETNJI DAN DO PODNE

Ostatak instalacije protekao je znatno mirnije, premda ne bez problema. Najviše me je namučio miš – priključio sam „najobičnijeg" *Genius*-a koji se u okviru svih komercijalnih programa ovoga sveta konfiguriše kao *Mouse System Mouse*, a u krajnje neverovatnoj situaciji da takve opcije nema, može da emulira i *Microsoft Mouse*. Do sada nisam video program koji ne bi podržavao bar jednog od ova dva miša, ali to samo znači da nisam video dovoljno programa. OS/2 ne pominje ni jednog ni drugog! Ostalo mi je samo da izaberem jednu od ponuđenih varijanti (mislim da se zvala *COM1 mouse*) i vidim šta će biti. Bilo je da se sistem ponašao kao da nemam miša, pa je trebalo izvršiti do-instalaciju koja, verovali ili ne, zahteva da se umetne **svaka** disketa između pete i petnaeste, za trenutak zavrti disk a onda zahteva sledeću bez obzira što sa tekuće nije ništa pročitano. Negde oko desete diskete pronađe se željeni drajver za miša, ali OS/2 svejedno zahteva umetanje i ostalih 5 (svaka čast onome ko je ovo izmislio, pitam se samo da li je IBM „kupio stručnjaka" koji je pisao instalacionu proceduru za *Logitech Modulu 2* ili za *3D Studio*). Miša sam, najzad, konfigurisao kao *Logitech Mouse* (što sasvim sigurno nije!), s tim što je sve vreme radio jako čudno – umesto desnog tastera koristio se srednji (?), a i kurzor se prilično skokovito kretao. No, radilo je.

Konfigurisanje video kartice prošlo je gotovo neprimetno – OS/2 je primetio da je u pitanju VGA, postavio kontrolno pitanje i dovršio instalaciju. Na žalost, Super VGA se tretira kao i VGA, ništa od mog omiljenog moda 800*600 osim za srećne vlasnike PS/2 računara. Ne sumnjam da će se uskoro pojaviti i Super VGA OS/2 drajveri za *Tseng*, *Trident* i ostale VGA kartice ali... za sada ih nema. Koliko sam primetio, vlasnici *Hercules* kartice ne mogu da instaliraju OS/2.

Čitava instalacija je, uz sve ove „stranputice" trajala dobra 2.5 sata – ako OS/2 instalirate više puta na sličnim konfiguracijama, računajte da je za svaku od njih potrebno otprilike 40 minuta, koliko minimalno traje unošenje odgovora na pitanja, umetanje svih disketa (poslednjih 5 sadrže drajvere za štampač, umeće se samo jedna od njih) i kopiranje njihovog sadržaja na hard disk. Nagrada za završen posao je efektni OS/2 ekran sa 7 ikona: *Start Here*, *Information*, *Master Help Index*, *Minimized Icon View*, *Templates*, *Shredder* i, naravno, *OS/2 System*; pojavljuju se i oznake uređaja na sistemu, u mom slučaju *Drive A* i *HP LaserJet III*. Sadržaj glavnog prozora (*OS/2 System*) me je, moram priznati, prilično iznenadio – u njemu sam našao samo *Productivity* (razni glupi programi poput kalendara, rokovnika i kalkulatora – bar da staviše kalkulator koji ima nešto osim četiri računske radnje!), *System Setup*, *Startup* (neka vrsta grafičkog AUTOEXEC.BAT-a), *Games* (na raspolaganju je loše urađen šah, dobro urađen (ima opciju za automatsku igru i varanje!) ali loše nacrtan pasijans, *Reversi* iz *Windows*-a i par drugih sitnica, recimo program koji nacrta mačku koja juri po ekranu i koju treba da „lovite" da biste uvežbali rad sa mišem) i najvažnije, *Command Prompts*.

NAJZAD ONO PRAVO

Otvaranjem ovog foldera dobija se prilika za pokretanje većeg broja poslova grupisanih u više tipova. „Osnovni" oblik je OS/2 *Full Screen*, komandni OS/2 mod kome je posvećen čitav ekran; ukoliko želite da pratite više



OS/2 procesa istovremeno, opredelite se za nekoliko *OS/2 Window* varijanti. Slično tome, na raspolaganju su *DOS Full Screen* i *DOS Windows* kao i *Windows 3.0* prozor koji je očito „podvrsta" DOS prozora u kome se izvršava specijalno napisani *Winodws 3.0*. Ukoliko u nekom trenutku pokrenete veći broj poslova, sa Ctrl Esc dobijete listu posredstvom koje možete da izaberete kom ćete se odnjih posvetiti; svi poslovi koji su ostali u pozadini se (ako eksplicitno ne zahtevate da bude drugačije) neprekidno izvršavaju, čak i ako se radi o nekom „prljavom" DOS programu: ni jednom mi se nije desilo da neki od takvih programa uspe da ispiše nešto van svog prozora ili da na bilo koji način omete druge programe.

Najviše me je, prirodno, zanimao OS/2 mod ali se u njemu, ako ćemo iskreno, ne može Bog zna šta videti. Komande su uglavnom identične DOS-ovim (promenjene su neke poruke, npr. XCOPY sada daje *Source files are being read* umesto DOS-ovog *Reading source files*) uz nekoliko dodataka koji se tiču *multitasking*-a, disketa se, na primer, formatira sa START FORMAT A: čime se startuje proces koji će se baviti formatiranjem disketa dok vi radite nešto drugo. Međutim, nisam imao nikakav pravi OS/2 program – uz operativni sistem se ne dobija ni C, ni bejzik, ni bilo šta što bi demonstriralo prave mogućnosti OS/2 tako da se sve svodi na istovremeno startovanje većeg broja DOS programa.

Iako je OS/2 komandni mod tekstualni, ne dopustite da miš „odluta" čim zatražite bilo kakav HELP ili čak pozovete priloženi editor zvani E, računar prelazi u grafički mod, iscrtava sve moguće prozore i najzad prozor posvećen željenoj operaciji – od tog trenutka do povratka u glavni prozor rad bez miša je nekomforan, a čitava operacija na iole sporijim konfiguracijama ume da potraje neprijatno dugo. Moram da kažem da takvim pristupom nisam baš oduševljen – navikao sam da radim sa tastaturom ili sa mišem, a ne ravnopravno sa jednim i drugim! Problem je posebno aktuelan prvih dana kada se često poziva HELP.

Što se DOS prozora tiče, veoma sam zadovoljan – velika većina stvari korektno radi. Postizao sam i neke efekte koji su me prijatno iznenadili – u jednom DOS prozoru, na primer, pokrenem drajver za YU slova i dobijem YU slova (po potrebi čak i umanjena!) koja ne utiču na ostatak sistema (OS/2 podržava YU slova u skladu sa standardnom *Microsoftovom* kodnom stranom, ali takvo rešenje je neupotrebljivo jer je neko, verovatno greškom, okrenuo slovo Š naopako). Ili, u jednom prozoru pokrenem OEMM kome dodelim onoliko *expanded* memorije koliko mu je potrebno i *Ventura* savršeno radi. DOS parametara ima dovoljno čak i za najambicioznija – njihovom promenom se svakom DOS prozoru može definisati okruženje koje obezbeđuje optimalan rad, uz angažovanja minimalnih sistemskih resursa!

Podešavanja prioriteta, međutim, ima neočekivano malo za jedan *multitasking* sistem – nisam, naime, primetio način da odredim prioritet nekog „pozadinskog" posla već se to mora prepustiti samom OS/2 koji prioritete dinamički raspoređuje prema nekim svojim internim kriterijumima. Sve to funkcioniše beskrajno bolje od *Windows*-a ili *DesqView*-a, ali se pri većem opterećenju (naročito ako se komunicira sa hardverom, npr. modemom) itekako primećuje čekanje na odziv u radnom prozoru, što nije baš prihvatljivo. Možda se na nižem nivou komunikacije sa OS/2 mogu podešavati prioriteti – iz komandnog moda je to nemoguće.

HVALA, ZA SADA NE

Što se problema tiče, nije da i njih nema, prvi i osnovni je u tome što ne funkcioniše baš sve. Ukoliko, na primer, uzmete DOS instalaciju *Windows*-a 3.0 i pokušate da je pokrenete iz DOS prozora, dobićete samo poruku o grešci – mora se koristiti *Windows 3.0* ugrađen u sam OS/2! U OS2 direktorijumu ćete pronaći i razne programe koji *patch*-uju razne DOS i (naročito) *Windows* programe tako da budu OS/2 kompatibilni – najprijatnije iznenađenje je svakako *WordPerfect for Windows* koji je, posle ove automatske ispravke, radio daleko pouzdanije nego u prisustvu „običnog" *Windows 3.0*. Problema ima i sa raznim uslužnim i keš programima (premda ovi poslednji više i nisu potrebni) koji pristupaju disku na nižem nivou – stari NU, na primer, možete pokrenuti samo sa NU /M pristupajući tako direktno disku, a ne njegovoj logičkoj strukturi; ne sumnjam da će sledeće verzije ovih programa biti HPFS kompatibilne. Primećuju se, najzad, i neke „mikro razlike" koje pokazuju da je čitav DOS zapravo ponovo napisan za potrebe OS/2: ukoliko, na primer, otkucate DIR PROG, biće prikazana samo (eventualna) datoteka PROG bez ekstenzije dok je na „pravom DOS-u" ova naredba bila ekvivalentna sa DIR PROG.*.

Mali haos je nastao posle instalacije *WordPerfect*-a 5.1 (DOS) – dok sam se držao normalne grafike, sve je bilo u redu ali kada sam instalirao drajver za Super VGA karticu i isprobao mod 1024*768, OS/2 je počeo da se „trese" – ako bih izašao iz grafičkog moda pa onda prešao u neki drugi task, sve je bilo (skoro) normalno, ali je promena taskova iz Super VGA grafičkog moda dovodila ekran u neko nedeterminisano stanje iz koga se teško izlazi, uglavnom nasumičnim udarcima po tastaturi i mišu.

Svi ti problemi su, međutim, prava sitnica prema onome što se desilo kada sam isključio računar – dobro znam da pre gašenja računara na svakom „pravom" operativnom sistemu treba najpre izvršiti neku manje ili više složenu proceduru, ali se dotična procedura nije pojavljivala ni u jednom meniju a računare je trebalo sklapati i ... off. Muka je nastala sutradan – posle uključenja računar je najpre krenuo sa CHKDSK-om, primetio problem kod datoteke SWAPPER.DAT (ta datoteka se, sasvim prirodno, u *multitasking* sistemu najčešće menja), počeo da ga ispravlja i ... *Internal error, System Stopped, Contact Your Vendor*. Novi pokušaj, ista stvar. Treći pokušaj, interna greška još pri „podizanju" OS/2. U normalnoj situaciji bih zaključio da nema izlaza i ponovo instalirao čitav OS/2, ali nemojte zaboraviti da bi za to trebalo opet rasklapati dva računara da se dođe do malog drajva, bio sam, dakle, sklon da učinim sve što mogu da to izbegnem. To „sve" je trajalo nekih 90 minuta, uključivao sam i isključivao keš na procesoru i disku, resetovao računar, pokušavao da prekinem CHKDSK... Upornost se uvek isplati, tako da je CHKDSK najzad „prošao", posle čega je interna greška nastupala pri crtanju glavnog ekrana, pa pri iscrtavanju ikona. Najzad je sve došlo na svoje mesto osim moga poverenje u pouzdanost OS/2 koje je (bojim se nepovratno) potrošeno – jedan operativni sistem koji drži do sebe jednostavno ne sme doživljavati ovolike interne havarije u jednoj ipak sasvim realnoj situaciji nenadanog gašenja računara. Sličnu havariju sam doživeo još jednom, ali bez ikakve „krivice" – samo sam otkucao NCD i sve se „smrzlo", računar sam morao da resetujem, a onda je trebalo proći kroz lavirint internih grešaka.

Čitaoca će, posle ovih iskustava, verovatno zanimati kako se sistem regularno isključuje. Jedna od onih stvari koje su logične kada ih znate ali kojih se nije lako setiti – treba dovesti miša na bilo koji prazan deo osnovnog ekrana (*Desktop*-a) i pritisnuti **desni** taster (da nisam, smatrajući da znam „osnovne stvari", odbio da pročitam *Tutorial*, može biti da bih znao da OS/2, za razliku od *Windows*-a, koristi oba tastera miša). Pojavljuje se meni u okviru koga je opcije *Shut Down*, a vredi primetiti i opciju *Lockup* pomoću koje „zaključavate sistem" – za nastavak rada mora se uneti izabrana lozinka ili isključiti i uključiti računar uz već opisane posledice. Tražeći podatke o isključenju računara, naišao sam i na stavku HELP-a koja se zove *REXX Information* što sam u „prvom prolazu" preveo kao „ovo se slobodno može preskočiti" – radi se, zapravo, o novom komandnom jeziku koji je proizašao iz DOS-ovog jezika za pisanje .BAT programa, ali uz brojna unapređenja koja obuhvataju kontrolne strukture, promenljive raznih tipova, prave potprograme sa argumentima i svašta drugo. Ukratko, jezik o kome će tokom sledećih godina dosta pisati!

Šta reći na kraju? OS/2 mi se kao ideja veoma dopao, prijatno sam iznenađen njegovom kompatibilnošću sa DOS-om i *Windows*-om kao i (sa korisničkog aspekta) idealno rešenim *multitasking*-om. Međutim, njegovom pouzdanošću nisam ni malo impresioniran, i ne verujem da bih mu mogao poveriti neke podatka koji su mi zaista bitni. Osim toga, sam OS/2 nema mnogo smisla dok se ne nabave barem C kompajler i par drugih alatki koje će koristiti njegov pravi 32-bitni mod. Ukratko, za sada ga na svoj računar ne stavljam, ali daću mu novu šansu kada ga malo „srede".

Korisna adresa

PP Izvor
11000 N Beograd, Drugi bulevar 113
Tel. (011) 136-754, 135-323, 134-245

LOTUS POD PROZOROM

U jednom od predhodnih brojeva „Računara" objavili smo kratke utiske o „Lotusovom" *spreadsheet*-u za *Windows*, *1-2-3/W*. Sada je na redu detaljnija zavirivanje „ispod poklopca".

Vladimir Stamenović

Želim da iskoristim još jednu priliku da pišem o *Windows* programima jer predviđam i želim totalnu propast *Windows*-a. Moja razmišljanja na tu temu su vezana za pojavu OS/2 verzije 2.0 i jedva čekam da ga se domognem.

Paket *Windows* je bio pravi softverski hit 1991/92. godine. Prodat je u desetak miliona kopija, izašla je i nova, poboljšana verzija *3.1*, a stotine proizvođača softvera izgiboše da naprave program za ovu platformu. „Lotus" je među prvima izašao na tržište sa programima za *Windows*, a tu pre svega mislim na prilično uspešni tekst procesor *Ami Pro*. *1-2-3 for Windows* je kasnio, iako da sada nije u situaciji da ugrozi *MS Excel* mada, prema nekim informacijama, njegova prodaja raste. Konkurencija se još uvećava pojavom *Excela* verzija *4.0*. Treba ovog leta očekivati odgovor iz „Lotusa".

ISTO KAO PRE...

Da se vratimo na *Lotus 1-2-3 for Windows* (*1-2-3/W*). Program se pojavio u prodaji pre gotovo godinu dana. Radi pod *Windows*-ima *3.0* ili *3.1* i sa DOS-om 3.11 ili višim. Preporučuje se računar sa minimalno 2 MB RAM-a, a na disku će zauzeti, nakon instalacije, oko 5 MB. Cena mu je 595 USD (server verzija 695 USD), a kod nas se može nabaviti kod firme JUSACO d.o.o. iz Novog Sada (021 395-414) ili Udruženje banaka Jugoslavije (011-682-588).

Za korisnike starijih verzija ili verzija za DOS, najinteresantnije je činjenica da je *1-2-3 for Windows* potpuno vertikalno kompatibilan po pitanju makroa, datoteka i rada sa tastaturom. Novi korisnici *1-2-3/W* neće imati mnogo muke oko učenja, pogotovu ako poznaju *Windows*. Stari korisnici kojih ima samo šesnaestak miliona, neće imati nikakvih problema, jer je pored *Windows* sistema menija zadržan i stari klasični (**Clasic**) meni u jednoj liniji iznad tabele. Dakle, na pritisak tastera „/" pojaviće se dobro poznati meni, sa komandama na koje ste već navikli. Isto tako, ako ste naučili na komande iz WYSIWYG-a, možete ih pozvati na stari način („:"). Meni je to dobro došlo, a verujem da će tako biti i sa ostalima koji budu rešili da rade sa *1-2-3/W*, jer je vreme za obuku svedeno na minimum, stečeno znanje korisnika je zaštićeno, a aplikacije rade bez ikakvih prevođenja i promena. Drugi „neklasični" meni je standardni *Windows* GUI meni. Stalno je prisutan na ekranu, a možete mu pristupiti preko Alt-taster kombinacije ili mišem. Sadrži sve uobičajene *Windows* komande kao što su isecanje (**Cut**), ubacivanje (**Paste**), štampanje itd.

U *1-2-3/W* su podržani svi formati datoteka iz prethodnih verzija (WK1, WK3, WR1, WKS, ALL, FMT i FM3), tako da se one mogu odmah učitati i koristiti. Podržan je takođe i trodimenzionalni sistem radnih tabela, kao u *1-2-3* verzije *3.1*, a za koji je ispitivanjem utvrđeno da je lakši za korišćenje od sistema jedne radne tabele u jednom prozoru. Dakle, u jednoj datoteci možete imati do 256 tabela, od kojih svaka ima 256 kolona i 8192 reda, a matematika kaže da je to 536 miliona celija. Moguće je povezivanje (linkovanje) više tabela u jednu kao i pristup zasebnim bazama podataka. Isto tako, tu su **Solver** i **Backsolver**, alati o kojima smo pisali, a koji omogućuju nalaženje maksimalnih i minimalnih vrednosti linearnih i nelinearnih funkcija i to unapred i u nazad. Takav sistem je prvobitno bio ugrađen u *1-2-3* za OS/2, a sada ga ima i *Excel* kao i, delimično, *Quattro Pro*. Naravno, zbog različitih platformi (DOS, *Windows* i OS/2), snaga ovih alata nije ista – hoću reći da sam pročitao da stvar mnogo bolje funkcioniše u *1-2-3* za OS/2.



Dvostruki interfejs *Lotus 1-2-3 for Windows*

Zanimljivo je i originalno rešenje sa ikonama (**Smart Icons**) koje je sada uobičajeno u *Windows* programima. Ono omogućuje da izgradite sopstveni sistem ikona pomoću kojih ćete znatno ubrzati rad, jer će ikone, kada ih aktivirate klikom miša, „odrađivati" unapred predviđene poslove. Čak možete dizajnirati sopstvene ikone koje će pozivati vaše makro procedure. Naravno, ne treba spominjati da ih možete pozicionirati bilo gde na ekranu.

...ALI I SASVIM DRUKČIJE

Nove mogućnosti programa podržane su samo kroz GUI menije. Tu su **File**, **Edit**, **Worksheet**, **Range**, **Graph**, **Data**, **Style**, **Tools**, **Window** i **Help** meniji. Kada radite sa grafikonima onda imate i **Chart**, **Draw**, **Layout** i **Rearange** menije. Neke komande na koje smo navikli u predhodnim verzijama nisu podržane u GUI meniju. Na primer, u **Range** meniju nema **Erase** komande, jer bi se ona poklapala sa **Clear** komandom iz **Edit** menija. Komande možete birati pre ili za vreme označevanja ćelija nad kojima će one imati efekta. To je prvo uvedeno u *1-2-3* za OS/2, a takva mogućnost postoji i u *Excelu*.

Poput *Toolbar*-a u *Excel*-u, u *1-2-3/W* postoje već pomenute **Smart Icons** palete koje sadrže komplet ikona pomoću kojih ćete ubrzati rad u tabeli. Možete dodavati i oduzimati ikone iz velikog asortimana gotovih ili kreirati sopstvene koje će pozivati vaše makro procedure. Osim toga, moguće je napraviti biblioteku različitih grupa ikona, tako da u svakom momentu rada (kreiranje tabele, grafika, makro procedure itd.) koristite onu paletu koja vam najviše odgovara. Takođe, kao što sam već rekao, možete podešavati izgled, veličinu i poziciju ikona na ekranu.

Veoma je dopadljiva mogućnost pozivanja makroa kroz ikone. Rad sa makroima je još unapređen kroz dodatni set komandi i kroz mogućnost „pamćenja" ručno izdatih komandi koje ćete zatim ubaciti u makro proceduru (*macro recording*). Taj posao se obavlja kroz **Transcript Window**, a *1-2-3/W* pamti do 512 karaktera, bilo da su oni sa tastature ili aktivirani mišem. U svakom trenutku se mogu videti u pomenutom prozoru, a pomoću **Copy** i **Paste** komandi ubacićete ih u tabelu. Sve postojeće makro procedure iz predhodnih verzija *1-2-3* rade normalno u *1-2-3/W*, a jedina razlika je što se pozivaju sa Ctrl-taster umesto uobičajene Alt-taster kombinacije. To je zbog toga što Alt-taster kombinacija u *Windows*-ima služi za brže biranje komandi iz menija. Makroi se mogu pozivati i preko RUN tastera (Alt-F3), kao u verzijama *2.2* i višim.

Nove makro komande se odnose na dodatne mogućnosti koje nude *Windows* u odnosu na DOS. Na primer, meni komande koje počinju sa „/", „:" ili {**MENU**} pozivaju klasični *1-2-3* meni. Nove makro reči {**ALT**}, {**MENUBAR**} i {**MB**} omogućuju pristup GUI menijima. Druge komande, kao što su {**EDIT-COPY**} i {**LINKCREATE**} omogućuju različite razmene podataka između aplikacija u *Windows* okruženju. Uspostavljanje veze sa eksternim bazama podataka ide pomoću novih komandi {**ROLLBACK**} i {**COMMIT**}, a komanda {**FILE-ERASE**} pojednostavljuje posao oko brisanja datoteke sa diska.

U *1-2-3/W* ćete naći originalne komande **Edit Quick Copy** i **Edit Move** pored uobičajenih **Cut**, **Copy** i **Paste** *Windows* komandi. Nove komande rade slično klasičnim *1-2-3* **Move** i **Copy** komandama, ubrzavaju posao oko premeštanja i kopiranja sadržaja ćelija, jer ne koriste **Clipboard**.

PREZENTACIJA PODATAKA

Za rad i prezentaciju podataka u tabeli značajno je da je u program uključen *Adobe Type Manager*, koji daje dobru oštrinu slova za bilo koji tip displeja i printera. Ugrađeno je i 13 *PostScript* skala fontova iste firme. Zgodna stvar je mogućnost pravljenja biblioteke fontova za rad na različitim printerima. Postoji još i masa drugih mogućnosti vezanih za zadebljana, kurziv, podvučena ili bilo koju drugu kombinaciju slova. Moguće je crtanje okvira oko svih ćelija ili oblasti, promena boja svih elemenata na displeju, senčenje u raznim intenzitetima itd. itd.

Predviđeno je povezivanje i razmena podataka između *1-2-3/W* i drugih *Windows* aplikacija preko takozvane DDE (*Dynamic Data Exchange*) mogućnosti. To znači da, ako uspostavimo takvu vezu, promena podataka u jed-

Nastavlja se na strani 37

PROROK U SVOME SELU

Microsoft je nedavno izbacio na tržište novu, dosta poboljšanu varziju svog poznatog spredšit programa. *MS Excel* 4.0 je najbolji tabelarni program za rad u *Windows* okruženju, a da li je dovoljno biti biti prorok u svome selu - to je već drugo pitanje.

Vladimir Stamenović

Ove godine *Excel* obeležava petu godišnjicu na PC računarima. Pojavio se 1987. godine i u to vreme je reklamiran kao „treća generacija" spredšit programa (prve dve su bile *VisiCalc* i *Lotus 1-2-3*). Microsoftov cilj je bio da istisne Lotus kao lidera sa tržišta ovakve vrste programa. Kompanija je dobro predvidela da će računari sa 80286 i tada novim 80386 procesorima preovladati i da će onda, *Excel* postati najbolji spredšit program na tržištu za taj hardver, kao što je to bio *Lotus 1-2-3* za 8086 računare. *Excel* je takođe bio prva ozbiljna aplikacija koju je napravio Microsoft za *Windows* okruženje, a isto tako i prvi program koji je za rad zahtevao AT računar. Međutim, uprkos nesumnjivom kvalitetu, *Lotus 1-2-3* je i dalje ostao najprodavaniji spredšit program na svetu, pa je Microsoft morao da obrati pažnju i na kompatibilnost *Excel*-a sa 1-2-3. U verziji 4.0, *Excel* je potpuno kompatibilan sa 1-2-3 verzija 2.01, a za novije korisnici 1-2-3 moraju da koriste *Macro Translation Assistant* ili da kompletno prerade svoje makro procedure.

CENA U HARDVERU

MS Excel for Windows verzija 4.0 se isporučuje sa estetski i korisnički unapređenim uputstvima za upotrebu, pa je sa njima lakše raditi nego u prethodnoj verziji. Obimni korisnički vodič je podeljen u dve knjige. *User's Guide 1* sadrži kompletna uputstva za pokretanje programa, sve o radnoj tabeli, bazi podataka, grafikonima i radu sa tastature. *User's Guide 2* se nastavlja na prvi deo, sa objašnjenjima o mogućim analizama u tabeli, razmeni podataka sa drugim programima, kao i o drugim podešavanjima programa. Raspoloživa funkcija su objašnjena u zasebnoj knjizi Function Reference. Posebno je, kao i u prethodnoj verziji, data knjiga *Q+E for Microsoft Excel User's Guide*. Za korisnike Lotusa 1-2-3 koji prelaze na *MS Excel* štampano je uputstvo za tu namenu – *Switching to Microsoft Excel from Lotus 1-2-3*. Na kraju, tu je i registraciona karta. Čudi me da nema skraćenog uputstva, koje je već uobičajeno u ozbiljnim programima, a najčešće se naziva „Quick Reference".

Program se isporučuje na pet disketa od 1.2 Mb. Instalacija je jednostavnija nego u prethodnoj varziji. Igrom slučaja, morao sam da izvršim instalaciju na 286 računaru, a pošto je poznato da *Windows*-i i ovakvi programi na tim računarima rade dosta sporo, u ovom članku neću obraćati pažnju na tu činjenicu. Čak i Microsoft preporučuje 386 SX sa 2Mb RAM-a kao minimalnu konfiguraciju. Elem, instalacija teče automatski, a na korisniku je da se opredeli da li će odmah uraditi kompletnu, delimičnu ili minimalnu instalaciju *Excel*-a. Ja sam uradio kompletnu i to me je koštalo oko 12 Mb prostora na disku. Ako hoćete dobar program i uz to u *Windows* okruženju, onda mora da se plaća na svakom koraku.

S PODACIMA TAMO-AMO

U ovoj verziji *Excel*-a ima dosta poboljšanja i izgleda da ovakav kvalitet i broj ugrađenih novih funkcija neće biti skoro dostignut od *Windows* konkurencija. Generalni stav kompanije da se rad u programu mora učiniti što lakšim primenjen je i u ovoj verziji *Excel*-a. Konkretno, to znači da su mnoge funkcije povađene iz **pull-down** menija i dijalog boksova i ubačene u „toolbar" i kao deo *Wizard* funkcija što se prvi put pojavilo u *MS Works for Windows*.

Značajna i glavna unapređenja se ogledaju u, generalno gledano, lakšem radu, boljim alatima za analize, prezentacije i izveštaje. Posebno bih naglasio veću podršku korisnicima Lotusovog 1-2-3.

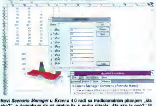
U samoj tabeli, da krenemo od nje, ima vrlo, vrlo lepih rešenja. I u prethodnoj verziji *Excel*-a smo imali sjajan interfejs, koji je sada još bolji. Mnogi poslovi su olakšani. Primera radi, ranije ste u zaglavlju tabele unosili po ćelijama redom K1 (prvi kvartal), K2... Sada je dovoljno da u jednu ćeliju upišete K1, a zatim strelicom „zakačite" donji desni ugao ćelije i osvetlite sve one u kojima će *Excel* automatski popuniti K2, K3... Stvar funkcioniše i sa datumima, mesecima ili bilo kojom jasnom serijom podataka. Još bolje, na sličan način se mogu prenositi i serije podataka koja sadrže neke međusobne relacije. Ovaj koncept se može koristiti i za brisanja podataka – „prevlačenjem" prazne preko punih ćelija. Izuzetno korisna stvar!

Velika novost je i mogućnost premeštanja (**Move**) podataka, odnosno ćelija po tabeli bez korišćenja **Cut** i **Paste** komandi. Sistem je vrlo jednostavan: osvetlite oblast ćelija i „vucite" je mišem na novu destinaciju. Na isti način, ali drzeći pritisnut Ctrl taster, možete uraditi i kopiranje (**Copy**). Sjajno!

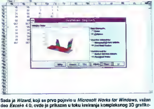
LAKOĆA FORMATIRANJA

Dosta funkcija iz klasičnog **pull-down** menija je prebačeno u **toolbar**, koji znaju to korisnici, puno pomaže u radu. Tu su sve značajnije funkcije, kao što su **Cut** i **Paste**, snimanje i učitavanje sa diska, grafika... a SUM ikoni dodata je i nova za oduzimanja. **Toolbar** je, po ugledu na *Lotus 1-2-3* i *Ami Pro SmartIcons*, moguće premestati po ekranu, kao i menjati njegovu veličinu.

Formatiranje tabela je znatno olakšano zbog novih **toolbar** funkcija. Izbor tipova slova, formatiranje brojeva, pozicioniranje teksta i brojeva unutar ćelije je samo deo onoga što se na vrlo jednostavan način može uraditi. Interesantno je da postoji 16 predefinisanih formata za celu tabelu koji na različite načine



Novi *Scenario Manager* u *Excel*-u 4.0 radi sa tradicionalnim pitanjem „šta ako?", a dozvoljava da ga pretvorite u seriju pitanja „šta ako je ovo?" ili „šta ako je ono?", na koja odgovara odmah i to na bazi različitih kriterijuma



Sada je *Wizard*, koji se prvo pojavio u *Microsoft Works for Windows*, važan deo *Excel*-a 4.0, ovde je prikazan u toku kreiranja kompleksnog 3D grafikona



Upotrebom „punjača" iz ugla ćelije, možete naterati *Excel* da automatski kompletira potrebne matematičke serije

pokazuju elemente u tabeli i oko nje. Izborom pravog formata možete sebi uštedeti puno rada, a podatke prezentirati na atraktivan način. Tu postoji i jedna zaista ekselentna sitnica. Naime, predefinisani format će automatski podesiti označenu oblast tako da svaka ćelija bude onakva kakva najviše odgovara podacima u njoj. Nema više nepotrebnog gubljenja vremena oko podešavanja širina kolona! Jedna od zgodnih novosti je i to što postoje posebni meniji koji nisu u vezi sa klasičnim, mada ponavljaju neke komande iz njih. Oni se pojavljuju u tabeli i to u zavisnosti od toga šta radite. Na primer, ako hoćete da napravite grafikon od podataka iz tabele, dobićete i taj posebni skraćeni meni koji će vam pomoći u poslu. Sve je to, naravno, jako kvalitetno i lepo urađeno.

Atraktivan je i deo programa koji je po prvi ugrađen u MS *Excel* 4.0, a nazvan je „*Wizard*" (čarobnjak). Pomoću njega je vrlo lako kreirati komplikovane tabele, jer vas kroz postupak vodi korak po korak. Interfejs je urađen sa puno mašte, svojim komandama podseća na kasetofon, a sve to zajedno je dosta funkcionalno.

RADNA SVESKA

Prema istraživanjima, većina korisnika upotrebljava samo oko 5% mogućnosti programa, a i tu ima problema. Da bi te osnovne stvari učinilo prihvatljivijim i lakšim, *Excel* sadrži neku vrstu „inteligencije". Primera radi, u *Excel*-u 3.0 je postojao takav problem sa **workspace** komandom. One je napravljena da bi u **workspace** datoteci omogućila čuvanje liste otvorenih dokumenata, izgleda prozora, uključujući i grafikone. No, **workspace** datoteka je čuvala samo te informacije bez pripadajućih dokumenata. Takođe, mogli ste da linkujete ćelije iz različitih datoteka, ali je i tu bilo problema, ako ste u linkovanoj tabeli promenili podatak, a niste je otvorili u isto vreme, veza se neće ostvariti kako je predviđeno.

U verziji 4.0, takve stvari su urađene dosta spretnije i, ne mogu da ne kažem, i dosta drugačije nego u **Lotusu 1-2-3** sa trodimenzionalnom tabelom. Sistem je nazvan **workbook** (radna sveska). Možete da uspostavite odnos među tabelama ili grupama datoteka tako da su one deo jedne datoteke. Povezaćete tabele u jednu i to će postati jedan **workbook** koga ćete tretirati kao jednu datoteku, s tim da možete dozvoliti da datoteke ostanu i kao individualne, odnosno neudružene u **workbook**. Zgodacija ovakvog sistema je u tome što se tabele koje čine **workbook** mogu fizički biti bilo gde (u uputstvu kažu – čak i na različitom serveru iz mreže). U takvom sistemu ima i problema. Recimo, ako u nekom izveštaju koristite podatke iz više datoteka, a da oni pri tom treba da budu sortirani.

Meni je **workbook** bio dosta zanimljiva stvar, jer prethodna rešenja sa čistim linkovanjem nisu dozvoljavala dovoljno komoditeta u radu. Ako već radite sa *Excel*-om, ovo je značajno unapređenje. Meni je trebalo malo vremena da to isprobam, i obzirom na okruženje, zadovoljen sam. Možda zbog navike, sklon sam da tvrdim da mi se rešenje iz *Lotusovog 1-2-3* verzija 3.1+ (nije za *Windows*) sa više tabela u jednoj datoteci više dopada. Interesantno je da se kod prevođenja višedimenzionalnih tabela (u jednoj datoteci) iz 1-2-3 u *Excel* 4.0, automatski pravi **workbook** koji ih povezuje kao zasebne tabele.

SCENARIO MANAGER

Interesantnih stvari ima i u delu koji se odnosi na **What-if** analize, za šta su ovakvi tabelarni programi idealni. O tome se pisalo u „Računarima" i korisnici koji to prate već bi trebalo da imaju predstavu o snazi raznih alata za takve analize. U *Excel*-u 4.0 ima jedna novost. Naime, možete, oko nekog problema, izgraditi više scenarija pomoću **scenario manager**-a, kako biste se lakše bavili analizom i tačnije utvrdili šta je za vas najpovoljnije. Obično se ispitivanje vrše sa istim podacima, a svaki scenario može biti u zasebnoj tabeli. Tim sistemom možete doći do različitih rezultata ispitivanja, od onih odličnih preko realnih, do krajnje loših. Preko **scenario manager**-a *Excel* je, najzad, postao značajna konkurencija, u ovom segmentu, Lotusovim 3D tabelama. Čini mi se odličnom mogućnošću da korisnik sebi olakša **What-If** analize.

Što se tiče grafičke prezentacije podataka, svima je poznato koliko ja ona bila kvalitetna u verziji 3.0. Možda je jedini nedostatak, ako se to tako može reći, bilo to što je trebalo dosta vremena da bi se grafikon dodatno obradio i izgledao vrhunski. U verziji 4.0 je i u ovom delu programa primenjena *Wizard* tehnologija. To za korisnika znači da možete napraviti grafikon i onda manipulisati s njim sve dok ne budete potpuno zadovoljni izgledom. Tek nakon toga ubacićete ga u tabelu. Posebne zanimacije je red sa 3D graficima, jer ih možete rotirati po volji sve dok svojim izgledom ne zadovolje namenu. Očekivao sam da će zbog konkurencije (*QuattroPro*) biti moguće dodati i ton uz grafiku, ali je takva mogućnost izostala. No, zato je prezentacija podataka koja se obično naziva slide show, dosta nalik na onu koja postoji u *QuattroPro* 3.0. Dakle, možete kreirati niz grafikona u različitim formatima, povezati ih u celinu i odrediti neki od brojnih načina na koji će se oni međusobno smenjivati tokom prezentacije. Čini mi se da nije predviđena mogućnost da se pokrene slide show bez *Excel*-a, direktno iz DOS-a ili *Windows*-a. No, moj je utisak da su sva rešenja koja se odnose na kreiranje i prezentaciju grafike urađena vrhunski.

Veliki napredak je postignut i u štampanju. Ugrađena je „fit-to-page" mogućnost koja važi i za štampače koji nisu PostScript. Možete po želji odrediti veličinu strane na kojoj štampate, a program će odraditi ostalo. Isto tako, ako imate kolor ekran, neće biti problema da skinete skrin na crno belom printeru.

Gejts i Kan... ...o konkurenciji

O NJIMA SVE NAJGORE

Bil Gejts (Bill Gates – osnivač Microsofta) i Filip Kan (Philipe Khan – osnivač Borlanda) govore o sebi i o – konkurenciji.

Nova verzija *Excel*-a je uključena u *Windows* tržište na bolji način nego ikada – kaže Bil Gejts, osnivač Majkrosofta. Nešto oko 1200 *Windows* aplikacija je proizvedeno prošle godine, a najmanje još 1200 se očekuje 1992 (mada je samo mali deo tih proizvoda iz spredšit familije). Svi oni žele da budu prisutni na *Windows* platformi, kao što je to bio 1-2-3 pod DOS-om.

Čudno je da je Lotus Development bio skoro godinu dana van igre sa svojim dugotrajućim obećanjem – *1-2-3 for Windows*. Čak i pre pojave *Excel*-a 4.0, Bil Gates je bio skeptičan prema konkurenciji oličenoj u Lotusu 1-2-3/W. „To je (misli na 1-2-3/W) najveći poklon koji smo ikada dobili. Neverovatno je da Lotus može prodavati takav proizvod. Gde su nove mogućnosti? Gde je brzina? To je tako sporo! Čudo je da uopšte prodaju 1-2-3/W"

U međuvremenu, osnivač Borlanda, Filipa Kan tvrdi da ga ne brine *Lotus 1-2-3 for Windows*, kao ni činjenica da da će se njegov *QuattroPro for Windows* pojaviti najranije šest meseci posle Lotusovog proizvoda. „Ne, mi ne brinemo zbog zakašnjenja – štos je biti najbolji, a ne prvi. Mogli smo poturiti Lotus na tržištu, ali to nije bila naša strategija. Ljudi će kupovati najbolji *Windows* spredšit. Naša istraživanja govore da korporacije čekaju sve dok ne mogu da uporede sva tri proizvoda (*1-2-3 for Windows*, *Microsoft Excel* i *QuattroPro for Windows*), a to je naša šansa. *QuattroPro for Windows* je za nas veoma važan proizvod, biće veoma važan i za industriju, jer će definisati nove standarde u spredšitovima"

U stvari, u februaru je *Newsbytes* agencija izjavila da je *QuattroPro for Windows* po bodovima i ocenama već ispred i *Excel*-a i 1-2-3/W. Na sajmu DEMO 92 u Indio, Kalifornije, u januaru, Borland je tvrdio da je QP/W čak i kao beta verzija bolji od konkurenata.

Borland je poredio *QuattroPro for Windows* sa *Lotus 1-2-3 for Windows* i *Microsoft Excel*-om 4.0 u tri kategorije. To su osnovni unos podataka i formule, analize i grafika i makroi. *QuattroPro* je bio bolji u prve dve kategorije, dok je u trećoj bilo nerešeno. Međutim, Borland nam još uvek nije ponudio da vidimo i testiramo program. Čekamo šansu...

PCW maj 1992

NAJBOLJI MEĐU WINDOWS-ima

Pred kraj, da se ponovo vratimo na komercijalne stvari. Kao što rekoh, i pored svih ovih kvaliteta koje ima *Excel*, Lotus 1-2-3 je i dalje spredšit broj jedan u svetu. Da bi mu na kraju došao glave, Microsoft je u *Excel* 4.0 ugradio i kompatibilnost sa komandama sa tastature iz „Lotus Clasic" menija, podršku svim Lotusovim formatima datoteka, uključujući i Impress (FM), kao i WK3 trodimenzionalne tabele. Osim toga, tu je Lotus Macro Interpreter koji direktno čita makro procedure iz 1-2-3 i ubacuje ih u *Excel* bez modifikacija. Sada se može reći, prema Microsoftu, da nema razloga da dosadašnji korisnici 1-2-3 ne pređu u klub *Excel*-a. Ja ću, ako to ikada poželim, prvo detaljno proveriti stvar.

Na kraju, moram reći da su moji prvi utisci u ovom prvom pregledu zaista povoljni. Svi delovi programa su brižljivo i izuzetno atraktivno urađeni, a najvažnije je da sve to zajedno odlično radi. Mada sam napomenuo da o brzini rada neću govoriti, moram da kažem da nisam ni tu imao nikakvih primedbi. Sve u svemu, MS *Excel* je odličan, svakako najbolji *Windows* program koji sam do sada video.

Korisna adresa

CET
11000 Beograd, Skadarska 45
Tel. (011) 343-043

SA MEKA NA PC

Može biti da je reč o akupom proizvodu, ali na bazi odnosa cena/performanse grafički paket firme „Adobe" može sasvim da se meri sa paketom *CorelDraw!* – a ima i mogućnost razmene datoteka između različitih platformi. Ovde vam donosimo prikaz beta-verzije ovog paketa.

Dugo je vremena bilo potrebno, ali najzad je tu – skoro. Početkom ove „godine „Adobe je najavio lansiranje *Illustrator*-a *4.03* za *Windows*. Ne, niste propustili verzije dva i tri, jer su one bile na raspolaganju samo za „Macintosh" računare. Ali pošto je nova verzija za *Windows* bogatija mogućnostima nego poslednja verzija (3.2) za „Maca", „Adobe" je odlučio da je označi većim brojem.

S JEDINICE NA ČETVORKU

Illustrator je bio prvi ozbiljni paket za crtanje za stone računare, i prvi paket koji su grafički umetnici zapravo mogli da koriste da bi postigli dobre rezultate. „Adobe" je utirao put na „Macu", ali je na PC-ju glavnu reč preuzeo *CorelDraw!* firme „Corel Systems", kome nije bilo ravna po kvalitetu od kada je predstavljen.

Sada „Adobe" uzvraća, a *Illustrator 4.0* je udarac usmeren pravo na *CorelDraw!*. Prva verzija *Illustratora* (1.0) na raspolaganju za PC bila je bazirana na *Windows*, ali bilo je to vreme pre pojave *Windows 3.0*. Sa dolaskom *Windows 3.0*, *Illustrator* je nadgrađen do verzije *1.1*, ali to nije bio neki sjajan softverski proizvod, što je izgleda smatrao i sam „Adobe", jer se nije potrudio čak ni da ga reklamira. Ove verzije je izdata jednostavno da se obezbedi kompatibilnost datoteka urađenih na „Macu" i na PC-ju. Za ostale stvari program je bio uglavnom neupotrebljiv. S obzirom na ono što smo videli u beta-verziji, nova verzija će doneti značajan napredak.

Ako niste radili sa „Adobeovim" *Illustrator*-om, verovatno ste koristili ili ste barem čuli za njihov *PostScript* – jezik za opisivanje stranica, u upotrebi na laserskim štampačima. Drugi veliki uspeh ove kompanije bio je ATM (*Adobe Type Manager*) koji stoji na raspolaganju za „Mac" i *Windows* okruženje i koji omogućava svakom štampaču koji je podržan u ovim operativnim sistemima da koristi *Type 1 PostScript* fontove.

Zajedno sa novom verzijom za *Windows*, *Illustrator* je sada na raspolaganju na tri različite platforme (druge dve su „Mac" i „NeXT"), pri čemu je ostvarena međusobna kompatibilnost njegovih datoteka. Na PC-ju ovaj program zahteva procesor 386 ili 486, a sam „Adobe" preporučuje 4 Mb RAM-a. Takođe će vam biti potreban i prostran hard-disk (beta-verzije se isporučuje na 10 HD 3.5-inčnih disketa). Na prvih šest disketa je sam *Illustrator*, dok su na ostalima prateći programi. Uz *Illustrator 4.0* dobijete ATM, *TypeAlign*, *Streamline* i separator boja. Miš će vam dobro doći.

INSTALACIJA PROGRAMA

Instalacioni program vam omogućava da izaberete koje od *Illustrator*-ovih programa želite da instalirate. Kada ste odlučili šta ćete da instalirate, možete dalje da podelite *Illustrator*-ov program na ostale komponente. Podrazumevano (default) se instalira veliki deo programa, što zauzima skoro 8.5 Mb prostora na disku. Ovo se može smanjiti minimalno na 2.5 Mb uklanjanjem „clip-art" i help datoteka, ali to, takođe, znači da ćete ostati i bez import i eksport filtera koji su mnogo značajniji za rad.

Illustrator je vektorski grafički paket, inače poznat kao objektno orijentisan. Slike (*images*) načinjene su od mnogo različitih objekata koji mogu biti pomerani, može im se menjati veličina, i uopšte može se sa njima generalno manipulisati. Ovo je suprotna koncepcija od one primenjene u grafičkom programu kao što je *Windows Paintbrush*, koji kao rezultat daje bit-mapirane slike – u osnovi jednu gomilu boja.

Mnogi popularni slikovni formati datoteka (*image file formats*) su bit-map tipa i ne mogu se učitati u vektorske grafičke pakete. Zato uz *Illustrator* dobijete *Streamline*, program za konverziju bit-map u vektorski format, sličan programu *CorelTrace* koji se dobija uz *CorelDraw!*, a tim što je *Streamline* malo savremeniji.



Ilustracije potencijala *Illustratora 4.0*: Slike jastoga nakon tri sata rada

Kada kliknete na ikonu *Illustrator*-a *4.0*, pojaviće se nakon kraće pauze početni ekran. Ovaj ekran sastoji se od uobičajenog *Windows* meni-bara, prazne bele stranice, pribora (*toolbox*) i statusne linije na dnu ekrana. Pribor sadrži 16 alatki koje su uvek vidljive, ali većina alatki ispod sebe imaju „sakrivene" alatke koje su sa njima u vezi. One postaju vidljive kada kliknete na neku od 16 alatki i pomerite miša na jednu stranu. Najviše pet alatki može biti „skriveno" ispod. Na raspolaganju u priboru ukupno imate 40 alatki čije funkcije su prilično šarolike. Većinu vremena radi se sa priborom, ali uprkos širokoj lepezi alatki nekim funkcijama programa može se pristupiti samo preko menija.

CRTAĆI ALATI

Prva od alatki u vertikali je, na primer, određivanje tipa pokazivača kojim vršimo izbora pri radu. Ima ih četiri i razlikuju se po referentnim tačkama (anchor points) koje selektuju: puna crna strelica služi za izbor celih objekata, dok bela strelica selektuje samo jednu referentnu tačku. Razlika između druga dva pokazivača je mnogo finija i oni služe za izbor objekata unutar grupe objekata.

Sledeći alat („ruka") služi za pomeranje prozora po slici koja je obično veća od dimenzija prozora. Uz pomoć ove alatke pomeranje po slici je često mnogo brže nego uz pomoć „klizača" na dnu i sa desne strane prozora, posebno kada se radi sa velikim uvećanjem koje efektivno sliku čini mnogo većom od radnog prostora. Uvećanje se kontroliše sa sledeća dva alata, oba u obliku stakla za uvećavanje: ono sa oznakom '+' uvećava, dok ono sa oznakom '-' umanjuje. Mogućnost koja je nova za *Illustrator 4.0* je tzv. „marquee zoom". Pomoću nje možete iscrtati pravougaonik oko nekog dela slike sa kojim želite da radite i taj deo slike ispuniće ekran.

MANIPULACIJA TEKSTOM

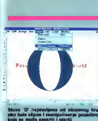
Mogućnosti manipulacije sa tekstom su impresivne. Za razliku od *Corel*-a koji od vas zahteva da tekst unesete u prozoru za dijalog, ovde se tekst crta (ispisuje) direktno na ekran, u odgovarajućoj veličini i sa odgovarajućim fontom. *Illustrator*-ove tri tekst-alatke imaju veoma različite funkcije. Normalna tekst-alatka je unos teksta u pravoj liniji, pri čemu pritisak na Enter započinje novi red. Tekst-po-liniji alatka radi tačno ono što joj ime govori: kada je izaberete i kliknete na bilo koju liniju na slici, možete unositi tekst koji će pratiti tu liniju, bez obzira da li je reč o kvadratu, krugu ili bilo kakvoj složenoj liniji povučenoj slobodnom rukom.

Postoji takođe i alatka koja vam omogućava da unesete tekst unutar nekog objekta (oblika), pri čemu zidovi objekta čine granicu teksta. Ako je tekst predugačak da bi bio uokviren jednim objektom, drugi objekti se mogu „povezati" sa originalom tako da teskt može da teče iz jednog objekta u drugi.

Sve ovo je prilično korisno, ali efekti sa tekstom postaju impresivni kada pokrenete „sestrinski" program – *TypeAlign*. On se koristi za savijenje, okretanje i uvrtanje teksta. Fina stvar ovde je da bez obzira koliko vi mnogo izmenili izgled teksta, to i dalje ostaje tekst koji se može izmeniti u neki drugi tekst. Iako je ova mogućnost nesumnjivo dobra, njena realizacija ne može da se meri sa kontrolom teksta u *CorelDraw!* u. Sa *Corel*-om, uvrnuti tekst se može menjati u svakom trenutku tokom dizajniranja kompletne slike, dok kod *TypeAlign*-a tekst mora biti kopiran u *clipboard* i zatim odatle „nalepljen" u *Illustrator*, gde postaje grafika.

SLOBODNOM RUKOM

Alat za crtanje slobodnom rukom omogućava vam crtanje, kao što je to uobičajeno, pritiskom na dugme miša i njegovim pomeranjem. Broj referentnih tačaka koje uokviruju oblik (shape) koji crtate zavisi od brzine kojom pomerate miša. Ako ga pomerate polako, dobićete veliki broj tačaka, a ako ga pomerate brzo, dobićete samo nekoliko. Oblici mogu biti otvoreni ili zatvoreni – ako se dva kraja linije dodi-



Slovo 'O' napravljeno od obojenog kruga oko bele elipse i manipulisanje pozadinom, koja se može prekriti i otkriti

ruju, oblik je zatvoren. Razlika između ova dva je bitna ukoliko oblike popunjavamo bojom, a posebno kada oblik crtan slobodnom rukom treba pretvoriti u neki drugi oblik jer tada on mora biti zatvoren.

Druga najčešće korišćena alatka je tzv. „pen tool". Pomoću nje se iscrtava prava linija između bilo koje dve tačke, pritiskom na dugme miša gde želimo da linija počne i da se završi. Međutim, ako ne otpustite pritisnuto dugme prava linija postaje Bezierova kriva. Na ovaj način mogu se napraviti nizovi pravih linija i Bezierovih krivih, što crtaču daje veliku kontrolu pri crtanju složenih slika.

Osnovniji oblici, kao što su pravougaonici ili elipse, dobijaju se pomoću skupa od šest alatki, od kojih su dva za elipse i četiri za pravougaonike. Za pravougaonike postoje četiri alatke jer mogu da imaju oštre ili zaobljene „ćoškove". U drugim paketima je obično samo jedan alat za crtanje svakog od ovih oblika, ali u *Illustrator*-u postoje dva, jer je definisanje oblika moguće početi od njegovog centra ili (što je češće) od njegovog gornjeg levog ugla.

Mada se definisanje oblika od njegovog centra čini logičnim, malo paketa ga implementira na taj način. Mada ovo kod vektorski baziranih paketa nije tako važno, jer se oblik može pomerati jednom kada je nacrtan, kod bit-map paketa trebalo bi da se omogući crtaču da postavi centar oblika tačno gde želi, a ne da nagađa gde treba prvi put da klikne na miša. Kvadrati i krugovi se u *Illustrator*-u lako crtaju upotrebom istih alata, samo što je potrebno držati pritisnut i Shift taster dok se formira oblik. Sledeći na dole ili gore, ili čak na desno (jer pribor sa alatom možete pomerati i postavljati gde hoćete) su alati za manipulisanje oblicima koji su nacrtani. Moguće ih je prevrtati, iskriviti, rotirati ili im promeniti veličinu. Postoje dve opcije za svaki alat: možete slobodnom rukom povlačiti oblik u željenoj meri, ili putem dijaloga možete uneti tačno koliko želite da izmenite taj oblik. Verzije ovih alata razlikujete tako što dijalog verzija ima '+' na svojoj ikoni.

MEŠANJE I SPAJANJE

Jedan od najimpresivnijih alata, i verovatno jedan od najmanje korisnih, je alat za mešanje. Možete „umešati" jedan oblik u drugi, jednu boju u drugu, jednu debljinu linije u drugu ili kombinaciju svega ovoga. Ipak, problemi postoje pri mešanju delova teksta. Čak i kada se tekst pretvori u objekte, a različita slova grupišu u jedan objekt, mešanje će raditi samo na jednom objektu (jednom slovu), jer iako su različiti objekti grupisani oni se i dalje tretiraju kao odvojeni. Ova mogućnost ovde nije tako napredna kao kod *CorelDraw!*-a koji može mešati bilo šta u bilo kom pravcu, ali ne treba ceniti grafički paket samo po toj osnovi.

Sledeća četiri alata služe za promenu referentnih tačaka. Kvadrat ima četiri referentne tačke, po jednu u svakom uglu, i povlačenje bilo koje od njih će izobličiti kvadrat. Ukoliko presečete stazu između bilo koje dve tačke pomoću makaza (scissors tool), možete pomerati delove staze bez izobličavanja ostatka kvadrata.

Postoje i alati za dodavanje i uklanjanje referentnih tačaka. Tako, ako dodate referentnu tačku na sredinu gornje stranice kvadrata, a zatim je odvučete na gore, dobićete nešto što podseća na „kućicu". Ako uklonite jednu referentnu tačku, staze se preformiraju i formiraju trougao ograničen sa tri preostale tačke.

Najzad, tu je i alat za dodavanje Bezierove krive ili za promenu postojeće referentne tačke na krivoj (primer za sve ovo su dati na slici).

Jednostavan, ali koristan alat je merač. Pritisnite dugme miša na početnoj i na krajnjoj tački bilo čega što želite da izmerite i pojaviće se dijalog u kome će vam biti prikazano rastojanje u jedinicama koje ste odabrali i ugao koji formira linija između te dve tačke.

CRTANJE GRAFIKONA

Sledeći alat, nazvan „charting tool", služi za kreiranje grafikona različitih formi, od onih u obliku torte (*pie-charts*), preko grafika rasejenja (*scatter graphs*) do histograma. Podaci za grafikone mogu biti uneti direktno u malu tabelu za unakrsna izračunavanja, ili mogu biti preneti iz bilo kojeg paketa koji podatke može snimiti kao tekst. Grafikoni mogu biti obogaćeni slikama, i za početak su vam obezbeđene izabrane slike pogodne za grafikone, a kasnije možete dodati i svoje ukoliko to želite.



Tekst se može konvertovati u konturu, a kontura popunjavati proizvoljnom bojom ili uzorkom

Nedostatak koji može prilično da smeta i koji verovatno nije posledica beta-verzije ovog paketa, je taj da je onemogućen prelazak između prozora pomoću Alt-Tab kombinacije tastera. Prinuđeni ste da minimizirate prozor ako želite da radite sa nekim drugim programom. Do ovoga je došlo zbog toga što je Alt taster upotrebljen za promenu dejstva izabranog alata. Na primer, pritisak na Alt taster dok pomerate objekt napraviće kopiju tog objekta.

Velika prednost *Illustrator*-a nad *CorelDraw!*-om je mogućnost rada u preliminarnom (*preview*) režimu i sa slikama u boji i sa crno-belim slikama, tako da se izmene mogu videti bez prebacivanja između „artwork" i „preview" režima. Oba režima su podržana, jer je režim za pregled mnogo sporiji od crtačkog (*artwork*), ali zato ćete uopšteno znati kako će nešto izgledati.

CorelDraw! ovo rešava tako što dozvoljava da se za jednu datoteku otvore dva prozora istovremeno, jedan u preliminarnom i jedan u crtačkom režimu. *Illustrator* ne dozvoljava da se jedna ista datoteka dva puta otvori, ali je zato moguće imati otvoreno više različitih datoteka u bilo kom trenutku, tako da se može raditi na nekoliko slika odjednom bez da se startuje *Illustrator* još neki put. U isto vreme može biti otvoreno onoliko datoteka koliko vam to vaša memorija dozvoljava i one se mogu posmatrati istovremeno u različitim prozorima, ako to želite.

U 16 MILIONA BOJA

Illustrator podržava 16-bitne i 24-bitne boje, što znači da možete raditi sa paletom od najviše 16.7 miliona boja ako imate adekvatnu grafičku karticu. Podržani su i popularniji kolor-komunikacioni sistemi, a među njima *Pantone*, *Focoltone*, *Toyo* i *Truematch*. Paket takođe podržava široku lepezu formata slike, od kojih može učitati EPS, CGM, DRW, CHT, PIC, PICT, TIFF i WPG, dok slike može snimati u BMP, CGM, PCX, PICT, TIFF i WPG formatu, kao i u nekim manje korišćenim formatima.

Illustrator 4.0 za *Windows* u sebi sadrži sve funkcije ranijih Mac verzija, a dodate su mu još neke. Čak su i rezervisani tasteri isti.

"Clip-art", kao takav, se ne isporučuje uz ovaj paket, već samo kolekcija monohromatskih simbola kao što su strelice i zvezde. Tu je, takođe, i kolekcija okvira koji variraju od najobičnijih, pa sve do preterano kitnjastih. Poddirektorijum sadrži izabrane uzorke koji mogu biti upotrebljeni umesto osnovnih boja za ispunjavanje oblika. Na ovim slike se vidi kako tekst može biti pretvoren u objekt, a zatim popunjen sa uzorkom. Bilo kakav grafički objekat može biti pretvoren u uzorak i upotrebljen kao ispuna. Primer sa slika generisan je uz pomoć obojenih pravougaonika. To je takođe i dobar primer kako tekst brzo može biti izmenjen da formira neki logo ili slova čudnog izgleda.

U BLAGOJ PREDNOSTI

Illustrator 4.0 je prilično lak za korišćenje, mada vas svakako neće obdariti umetničkim talentom ukoliko ga već sami nemate. Nema potrebe da vam kažemo da mi nismo nacrtali sliku onog jastoga, ali vam ona može dati dobru ideju o potencijalima ovog grafičkog paketa. U pravim rukama *Illustrator 4.0* je jedan od najmoćnijih paketa za crtanje na tržištu. Dobro stoji u odnosu na jaku konkurenciju koju mu čine *CorelDraw!*, kao i relativno novi *Harvard Draw*. Glavna prednost koju „Adobe" ima je da je *Illustrator* proizvod za više platformi.

U okruženjima gde se koriste i „Macovi" i PC-ji, a možda u manjem obimu i NeXT mašine, *Illustrator* je u jasnoj prednosti zbog razmenljivosti datoteka. Ali ne dobija se jeftino. Očekuje se da će kompletan paket, zajedno sa *TypeAlign*-om, *Type Manager*-om i *Streamline*-om, plus izborom od 40 ATM fontova, koštati oko 565 funti. Ovo može izgledati skupo, to jeste skupo, ali u odnosu na performanse programa moglo bi se reći da se radi o pravoj ceni.

Biće interesantno videti da li će „Corel Systems" i SPC uzvratiti sa verzijama svojih paketa na drugim platformama. Dok oni to ne učine, *Illustrator* vodi.

Prema *Personal Computer World*
Priredio: *Kragl Aleksandar*

LJUBAZNI POSREDNIK

Programi za arhiviranje postali su sastavni deo softverskog „arsenala" svakog korisnika računara. Ali oni imaju i jednu nezgodnu osobinu – sva komunikacija se obavlja preko komandne linije prepune raznih parametara. U novije vreme počeli su da se pojavljuju razni univerzalni ili specijalizovani programi kojima je jedina namena posredovanje između korisnika i „glavnog" programa. Jedan od najpopularnijih među njima je SHEZ.

Jovan Selajić

Korišćenje arhivera je postalo navika, a i potreba, velikom broju korisnika računara. Verovatno i vi na svom hard disku i disketama imate gomilu spakovanih datoteka i često vam je potrebno da pronađete nešto među njima. Ili ponekad dobijete disketu punu novih programa (naravno arhiviranih) i želite da na brzinu vidite čega sve tu ima i pretstate dokumentaciju za svaki od njih. U takvim slučajevima pozivanje arhivera sa raznim njihovim opcijama i parametrima iz DOS-a može biti vrlo mukotrpan posao.

SHEZ je pravljen da vaš poštedi tih muka. Sve moguće operacije sa arhivama koje vam padnu napamet pomoću njega možete raditi brzo i lako preko menija ili kombinacija tastera. To uključuje kreiranje arhiva, pregled njihovog sadržaja, konverziju među raznim formatima, pravljenje samoraspakujućih arhiva, zaštitu od virusa, rad sa pojedinačnim spakovanim datotekama (njihov pregled, izvršavanje)... *SHEZ* je tzv. *shell* program i može se opisati kao neka vrsta *PC Tools*-a za arhive. *SHEZ* nije program koji u sebi objedinjuje sve arhivere – on je jednostavno samo ljubazni posrednik između vas i njih.

ZA POČETAK

SHEZ se distribuira u obliku ZIP arhive duge oko 190K, od čega se posle raspakivanja dobije oko 450K datoteka. Tu se, uz glavni program, nalazi još nekoliko pomoćnih programa, razna dokumentacija i PIF i ICO datoteka za rad pod *MS Windows* (gde *SHEZ* radi samo u DOS prozoru, prava *Windows* verzija se još ne najavljuje). Preporučujemo vam da otpakujete celu arhivu i pogledate šta sadrži, a da glavni program SHEZ.EXE (a možete i help datoteku SHEZ.HLP, mada ona nije neophodna) iskopirate u neki direktorijum u kome držite druge uslužne programe, na primer C:\UTILITY. Ukoliko želite da koristite i opciju za traženje stringova unutar datoteka, prekopirajte i SGREP.COM, program koji služi navedenoj svrsi.

Za početak će vam biti neophodan još i konfiguracioni program SHEZCFG.COM, a preporučujemo vam da obavezno odštampate SHEZ.REF datoteku koja predstavlja kratki podsetnik (*hot key reference*) za sve komande *SHEZ*-a. Pravljen je za gusto štampanje (132 karaktera po redu, 8 linija po inču), zato prvo u DOS-u uradite jedno

MODE PRN: 132, 8 pa
COPY SHEZ.REF PRN

Glavna dokumentacija sadrži oko 150K teksta (sa ubačenim kontrolnim kodovima za štampač) ili 67 strana posle štampanja. Naš savet je da ipak poštedite svoj štampač ovolikog napora, jer vam dokumentacija neće biti preterano potrebna kad se malo naviknete na rad sa programom.

Uz sam *SHEZ* biće vam potrebni još neki dodatni programi koje verovatno svakodnevno koristite – jedan tekst editor, program za pregled datoteka (*file browser*) i program za kontrolu prisustva virusa u datotekama. Za *file browser* i autor *SHEZ*-a i mi vam preporučujemo LIST, a za viruse verovatno već imate *McAfee SCAN*. Naravno, potreban vam je i bar jedan program za kompresiju datoteka – arhiver. *SHEZ* trenutno podržava sledeće arhivere i njihove formate zapisa datoteka: PKZIP, ARJ, LZH, LHA, PKPAK (PKARC), PAK i ZOO.

KONFIGURISANJE

Konfigurisanje *SHEZ*-a se ne bi mogla nazvati jednostavnim. Potrebno je startovati SHEZCFG i odgovoriti na čak (najmanje) pedesetak pitanja. Glavni meni instalacionog programa sadrži devet opcija, od kojih su vam za početak bitne „Programs setup" i „Aditional setup information".

„Programs setup" je namenjen unosu osnovnih informacija o programima koji su uz *SHEZ* neophodni za rad. Treba da odlučite koje ćete arhivere koristiti, kazati gde se oni nalaze na hard disku i koja je njihova verzija, zatim idu slična pitanja za programe koje ćete koristiti za pregled datoteka i proveru od virusa. Podrazumevane (default) vrednosti su dobro izabrane, jedino vam skrećemo pažnju da za rad sa ARC datotekama umesto ponuđenih *PKpak & PKunpak* stavite (kod nas daleko rasprostranjenije) *PKarc & PKxarc*. Ukoliko se arhiveri koje koristite nalaze u direktorijumu koji je definisan u PATH-u, nije potrebno navoditi punu stazu do njih, već samo njihovo ime i ekstenziju.

U „Aditional setup information" vas očekuju dodatna pitanja vezana za privremene direktorijume koji će *SHEZ* koristiti za svoje potrebe tokom rada (naš savet: stavite isti direktorijum koji ste u DOS-u definisali sa SET TEMP), način prikazivanja na ekranu, dodatna pitanja za konfigurisanje arhivera, korišćenje memorije, upotrebu miša itd. Da bi ste poštedeli sebe muke oko smišljanja odgovora (na sreću i ovde sa dobro postavljenim podrazumevanim vrednostima), ovaj deo setapa možete da ostavite za neki drugi put, kad se malo naviknete na rad sa samim programom.

Ostale opcije se odnose na podešavanje boja, registraciju i proveru integriteta programa, i za početak vam nisu bitne (početne boje su sasvim pristojne), sem naravno „Save configuration info", kojom snimate izabranu konfiguraciju u SHEZ.EXE.

Registracija se vrši kada od proizvođača dobijete svoj serijski broj (naravno, posle uplate iznosa registracije) izborom opcije „Registration number instalation". U ranijim verzijama su „radili" i brojevi poput 1111... i 3333..., ali od ove verzije to više ne važi. Neregistrovana verzija je potpuno identična registrovanoj – razlika je jedino u prozoru sa uvodnom „shareware" porukom koji se pojavljuje prilikom svakog startovanja.

Instalacioni program ima i jednu manu i jedan (mali) bug. Mana je što po odabiranju određene opcije nikako ne možete prekinuti proces (uobičajenim ESC), već morate proći kroz sva potpitanja, što se na kraju svodi na besomučno lupanje po Enter i željno iščekivanje pojave uvodnog ekrana. Bug se ogleda u malo poremećenim bojama i senkama na ekranu posle pojave prozora koji ukazuje na greške korisnika tokom instalacije.

ZADAVANJE KOMANDI

Sve komande se zadaju preko kombinacija tastera (*hot key*) – Ctrl, Alt, Shift, funkcijskih i „običnih". Pomoć (spisak tastera sa njihovim funkcijama) se dobija uobičajenim F1, ili se može naći u odštampanom *hot key reference*. Moguće je zadavanje komandi i preko menija, koji se dobija na vrhu ekrana pritiskom na / (kosa crta, „slash").

Meniji su izgleda u program dodati naknadno i ne uklapaju se baš u autorovu koncepciju jednostavnosti – izabiranjem neke od opcija posle obavljene operacije ne vraćate se nazad u meni, već u početni *hot key* mod. Ukoliko tada ponovo pozovete meni, nećete se naći u onoj opciji koju ste poslednju koristili, već opet u početnoj.

Rad sa mišem je podržan. Konfiguracioni program vas pita da li želite da koristite ugrađenu podršku za to, ali smo primetili da vaš odgovor ovde ne služi ničemu (bar kod našeg „Microsoft" miša) – ukoliko odgovorite potvrdno, svejedno ćete posle morati pre rada da učitate drajver za glodara, a ukoliko stavite da ne želite da ga koristite, opet će *SHEZ* pri startovanju prepoznati učitani drajver i aktivirati podršku za miša.

Lična karta

Program: *SHEZ – The Compression Companion*
Verzija: 7.1, 12. April 1992.
Namena: Rad sa arhiviranim datotekama
Zauzet prostor na disku: 200–450K
Način distribucije: *shareware* (ASP), registracija 39 USD, neregistrovana verzija je takođe potpuno funkcionalna
Proizvođač: *California Software Design, James Derr, P.O. Box 15248, Santa Rosa, Ca. 95402, tel. (707) 575-9868*
Sezam: *SHEZ* možete naći u **\ibmpc\archiver** direktorijumu kao **shez71.zip**



Tipičan izgled ekrana pri radu sa celim arhivama..



... i datotekama unutar njih

Pri korišćenju miša, levi taster igra istu ulogu kao i Enter na tastaturi a desni kao ESC. Dvostruki pritisak (*double click*) na levi taster (ili običan pritisak na srednji ukoliko koristite miša sa tri tastera) aktiviraće na sredini ekrana poseban "mouse" meni sa nekoliko najčešće korišćenih opcija, u kome je prva opcija pozivanja glavnog menija.

Postoje dva osnovna režima rada – sa celim arhivama i datotekama unutar njih. Na **slikama 1** i **2** možete videti tipičan izgled ekrana pri radu u svakom od njih.

RAD SA CELIM ARHIVAMA...

Prilikom startovanja *SHEZ*-a, prvo ćete se naći u modu za rad sa celim arhivama. Sa leve strane ekrana je prozor sa prikazanim sadržajem direktorijuma (*file window*), dok se desna strana uglavnom ne koristi. Pritiskom na TAB se i sa desne strane može dobiti drugi prozor sa listom datoteka i direktorijuma, ali se nad njima mogu vršiti samo operacije kao i sa svim datotekama u DOS-u – kopiranje, pregledanje, brisanje, premeštanje, promena imena. Iste operacije se mogu obavljati i u glavnom prozoru. Možete izabrati da se prikazuju sve datoteke (*.*) ili samo arhive ($.$), to jest one za ekstenzijama ZIP, ARJ, LZH i ostalim definisanim u setapu.

Između datoteka krećete se kurzorskim tasterima (ili mišem), a prelazite u neki poddirektorijum pritiskom na ENTER kada ste pozicionirani na njega. Tako se i biranjem .. vraćate u prethodni direktorijum (CD ..), ali istu ulogu ima i taster ESC. Ovo je malo nelogično, a tek kada ste u osnovnom katalogu ESC služi očekivanoj svrsi – za izlazak iz programa (iz programa se može izaći i pritiskom na Alt-Q). Sa F2 menjamo aktivni direktorijum u bilo koji drugi na disku; njihovo stablo se može prikazati i kao u NCD-u.

I ovde postoje dva osnovna načina rada, to jest odabiranja arhiva nad kojima će se izvesti željene komande – rad sa trenutno izabranom, odnosno inverzno prikazanom datotekom (*highlighted*) i sa markiranim, to jest izabranim datotekama (*tagged*). U skladu sa tim (mada ne previše dosledno) dodeljene su i funkcije tasterima – markiranim datotekama su namenjene SHIFT kombinacije, a inverznim «CTRL» ili «ALT». Markiranje se vrši pritiskom na «Space» (ima i drugih kombinacija), a pored takvih fajlova se ispiše marker u obliku znaka za koren.

Operacije nad arhivama obuhvataju pravljenje samoraspakujućih (*self extracting*) datoteka, dodavanje/menjanje komentara, konverziju između raznih formata arhiva i provera prisustva virusa. Rad je izuzetno jednostavan – na primer od inverzne arhive dobićemo samoraspakujuću (.EXE) jednostavnim pritiskom na «F8». Moguće je i kreiranje novih arhiva od željenih datoteka.

Konverzija u druge formate je operacija koju za većinu arhivera morate ručno da obavljate. Uz ARJ se dobija odličan REARJ koji radi prebacivanje iz bilo kog u ARJ format, ali obrnut proces nije moguć. Nama je baš nedavno zatrebalo da neke arhive iz ARJ prebacimo nazad u ZIP format (tek posle REARJ smo primetili da instalacija jednog programa zahteva baš originalni ZIP) – *SHEZ* nam je tu uštedeo dosta muka.

... I DATOTEKAMA U NJIMA

Odabiranjem arhiva sa ENTER prelazimo u režim za rad sa spakovanim datotekama. Dobijamo njihov spisak (**slika 2**) sa osnovnim informacijama – imenom i ekstenzijom, kompresovanom dužinom, datumom i vremenom kreiranja, originalnom dužinom, metodom kompresije koju je koristio arhiver i stazu ukoliko su spakovani i poddirektorijumi.

Prilikom prikaza sadržaja arhiva, *SHEZ* još uvek ne poziva arhivere (to jest dearhivere tamo gde su oni zaseban program), već ima u sebi ugrađene funkcije za čitanje sadržaja svih podržanih arhivskih formata.

I u ovom režimu razlikujemo rad sa inverznom i markiranim datotekama. Neke komande se prenesene iz prethodnog režima, a neke su specifične samo za rad sa spakovanim datotekama.

Dodatnim pritiskom na ENTER pozvaće se program za prikazivanje datoteka (*file browser*) i biće vam omogućen pregled inverzno prikazanog fajla. U slučaju da je reč o arhivi unutar glavne arhive, biće opet aktiviran režim za rad sa pojedinačnim datotekama ovog puta sa podarhivom.

Na ovom primeru najbolje se vidi princip povezivanja SHEZ-a sa ostalim eksternim programima. Pritiskom na ENTER se najpre poziva arhiver sa argumentima za raspakivanje datoteke u privremeni direktorijum (tok ove operacije se vidi u posebnom DOS prozoru), a zatim i program za pregled sa raspakovanom datotekom kao argumentom. Sada dolazimo na dobro uređeno manipulisanje sa memorijom zbog koga je bilo toliko pitanja prilikom instalacije. *SHEZ* može prilikom pozivanja eksternih programa ili privremenog izlaska u DOS da veći deo svog koda premesti (*swap*) u EMS ili XMS memoriju ili na hard disk. Tako se potrošnja memorije prilikom DOS shell-a (koji zatreba ponekad) ili pozivanja programa sa više od 200K (kolika je sa početno podrazumevanim vrednostima) može svesti na manje od 30K. Ovo nije tako beznačajno kao što možda nekome izgleda – na primer ARJ arhiver zahteva slobodnih 290K za rad, što uz *SHEZ*-ove početne zahteve penje na više od 500K.

Vratimo se opet radu sa pojedinačnim datotekama unutar arhiva. Uz pregled omogućeno je i njihovo editovanje, pretraživanje, startovanje spakovanih programa, dodavanje komentara, zaštita šifrom, raspakivanje pojedinih fajlova ili cele arhive.

U *SHEZ*-u je moguće i dodeljivanje često korišćenih kombinacija tastera makroima. Ovo je urađeno na uobičajen način, kao i kod programa sa sličnim mogućnostima – opcijama za definisanje makroa, njihovo izvršavanje, snimanje i učitavanje. Ovim kombinacijama su dodeljene uglavnom kombinacije tastera «Alt broj», a operacije izvršene mišem ne uzimaju se u obzir.

(KAD) NEMA BOLJEG

SHEZ dosta olakšava posao, ali nije savršen. Stvar koja nam najviše nedostaje je nemogućnost popravljanja oštećenih arhiva. Neke operacije bi se verovatno mogle pojednostaviti, još nešto dodati (na primer podrška za brojne nestandardne opcije arhivera), možda bi se čak mogli ugraditi i kompletni algoritmi za kompresiju/dekompresiju pojedinih formata zapisa. Ali to je verovatno previše zahteva za autora programa (ipak je u pitanju samo jedan čovek) a verovatno i suviše daleko od njegove početne ideje.

SHEZ je dobar, jer je jedini. Nismo videli da se neki drugi dovoljno dobro urađen program ove namene može naći na ovim prostorima, a, iskreno, nismo se ni trudili da tražimo bolji. Ovaj ima ono što nam je potrebno, jednostavan je za upotrebu, nove verzije se redovno objavljuju. Pojava nekog konkurenta bi u svakom slučaju dobrodošla, naročito za sve veće *Windows* tržište, gde bi sve ovo trebalo i da bude veoma lepo. U svakom slučaju, ako do sada još niste koristili *SHEZ*, krajnje je vreme da počnete.

POSLE DOS-a

4DOS se pojavio u vreme suverene „vladavine" MS-DOS-a 3.30 kao odgovor na njegove brojne slabosti i nedostatke. U međuvremenu je u oblasti operativnih sistema i komandnih procesora došlo do dramatičnih promena (DOS 5.0, *Windows 3.1*, *OS/2 2.0*). Može li 4DOS da bude interesantan i u ovim novim uslovima?

Viktor Ohuljen

Kada sam u *Računarima* broj 68 (januar 1991) napisao nekoliko rečenica o 4DOS-u ideja mi je bila da obavestim korisnike PC računara da je odzvonilo vladavini COMMAND.COM-a i sličnih nefleksibilnih i nedovoljno user-friendly tumača naredbi. U tom trenutku sam bio beskrajno odan korisnik 4DOS-a, istina, tada još nisam koristio *Windows*-e i nije bilo „Microsoft" DOS-a 5.0. U međuvremenu su se stvari značajno izmenile. Naime, sa DESQView-a sam prešao na *Windows*-e, a i pojavio se MS-DOS 5.0.

ZAŠTO JE NASTAO...

U vreme pojave 4DOS-a nastali su i programi koji su omogućavali editovanje komandne linije i „vraćanje" prethodno zadatih naredbi. Takođe je u to vreme bilo nemoguće smanjiti otisak koji COMMAND.COM ostavlja u konvencionalnoj memoriji. Nedostatak na koji su se mnogi korisnici žalili bila je i nemogućnost prepravljanja *path*-a u toku rada, osim ponovnog zadavanja cele okolinske promenljive „*path*". Rad sa fajlovima u okviru *batch* datoteka je bio veoma ograničen i neudoban, a za svaki ozbiljniji posao moralo se obraćati nekom ozbiljnijem alatu, pa i zasukati rukave da bi se nešto napisalo na C-u.

To je bila situacija sa MS-DOS-om 3.30 kada se pojavio 4DOS i pobrao veliku slavu. Mnogi od nas su se „preobratili" i probavši ga jednom postali njegovi „ovisnici", prezirući svaki povratak na „obični" COMMAND.COM. U tom trenutku on je nudio sve što je gore navedeno kao mana MS-DOS-a i ponešto više. No o svemu tome smo pisali u tekstu u broju 68 prema verziji 3.xx. U međuvremenu se pojavila verzija 4.0, ali i novi MS-DOS 5.0 i stvari su se značajno izmenile.

...U ŠTA SE RAZVIO

Od 4DOS verzije 4.0 smo očekivali značajna poboljšanja proizvoda koji smo cenili, ali se sve svelo na nekoliko pojednostavljenja korišćenja, ali ne i na značajne novine. Naime, prvo sa čime se susrećamo jeste to da se sada, umesto podešavanja svega oko 4DOS-a u okviru kriptičnih prekidača smeštenih na jednoj liniji, sve to čini u okviru fajla 4dos.ini nizom prekidača, svaki na svojoj liniji, sa jasnim i prirodnim imenima. Dakle, umesto

```
SHELL=C:\...\4DOS.COM \@ c:\...\4dos-cmd.opt
```

gde je datoteka 4dos-cmd.opt recimo od jedne linije

```
/L C:\4DOS /S DG.\ /E:1024U /A 4096 /H:3072 /U/P wind
```

i sve to postaje

```
SHELL=C:\...\4DOS.COM @ c:\...\4dos.ini wind
```

gde je datoteka 4dos.ini recimo

```
Environment=1024
Alias=4096
History=3072
UMBLoad=Yes
UMBEnvironment=Yes
Swapping=c:\...\swap4dos\,XMS
StartPath=c:\4dos
AutoExecPath=c:\
HelpPath=c:\4dos...
```

Jeste preglednije i lakše za rad, ali nije ništa revolucionarno – bila je prva reakcija na tu promenu.

Nepravedno je uvek očekivati čuda od nekoga ko vam to jednom priušti. Tako je i sa 4DOS-om. Što se tiče editovanja komandne linije, stvari su se poboljšale ispravljanjem nekoliko problema na koje su korisnici ukazivali, ali na koje je malo njih nailazilo, i uvedena je mogućnost da dobijemo prozor sa spiskom svih prethodno zadatih naredbi i da jednostavnim pomeranjem biramo koju ćemo ponoviti. Na žalost, naredba izabrana iz tog prozora mora „ići" onakva kakva je – ne može se prepravljati. Istina, korišćenjem mogućnosti prizivanja prethodno zadatih naredbi samo navođenjem nekoliko njihovih prvih slova, rad postaje komforniji.

Novina i nema baš previše, ali sve je lepo pročišćeno i sređeno i radi skoro besprekorno. Probao sam novu verziju na mnogim mašinama i kao osnovnu školjku i kao program koji sam jednostavno pozivao sa naredbene linije i svuda je sve radilo kako treba – bez ikakvog gunđanja i problema.

Što se tiče problema i pitanja, njih zaista nema puno, o čemu svedoče dva mesta gde se pitanja uvek mogu postaviti i gde nikada nije dosadno sa paketima koji imaju bagova. Jedno od tih mesta je nezaobilazni *Sezam* i cela tema date 4DOS-u. Bilo je dosta interesovanja oko nove verzije jer, moramo priznati, kasnila je nekoliko nedelja, ali nakon toga sve je nekako utihnulo i svi su, čini se, zadovoljni. Drugo mesto je 4DOS-List na Bitnet-u, koju je pokrenula firma koja proizvodi 4DOS, „JP Software Inc.", i gde se takođe, nakon nekoliko problema pri radu sa mrežama, sve brzo smirilo. To samo govori o tome da nova verzija nije donela ništa revolucionarno, ali je pročišćena od grešaka i sasvim zrela kao proizvod.

> **Lična karta:**
>
> **Namena:** tumač naredbi
> **Verzija:** 4.0
> **Verzija DOS-a:** 2.0 ili veće
> **Hardver:** bilo kakav PC-kompatibilni računar sa odgovarajućim proširenjima
> **Prostor na disku:** zajedno sa help-om oko 700 KB (sa guided tour-om)
> **Dokumentacija:** uputstvo u fajlu na disketi i knjižica sa 180-tak strana
> **Proizvođač:** *J.P. Software, P.O.Box 1470, East Arlington, MA 02174, U.S.A. Telefon (617)646-3975, FAX (617)646 0904, BBS (Channel 1) (617)354-8873*
> **Trenutna preporučena cena (i za mail-order):** 70-tak US \$
> Gde se kao *shareware* može naći:
> BIX conference "ibm dos/4dos" ID "trewson"
> COMPUSERVE Internet Bitnet (general) „75300 210\@compuserve.com" (technical support) „75300 1215\@compuserve.com"
> TRICKLE $<$msdos 4dos$>$
> *Sezam* \IBMPC\SYSTEM

4DOS vs MS DOS 5.0

Kako testirati 4DOS u poređenju sa MS-DOS-om 5.0? To je pitanje koje sam sebi postavio u trenutku kada me je redakcija „Računara" nedavno pozvala da napišem novi prikaz ovog proizvoda. Prva ideja koja mi se nametnula i koja se pokazala kao pravilna bila je da prekinem korišćenje 4DOS-a kao osnovne školjke i probam rad sa MS-DOS 5.0 COMMAND.COM-om. Jasno, ova odluka nije bila nimalo laka.

Prvi problem se javio zato što 4DOS ima svoje posebne *batch* datoteke sa prezimenom btm (*batch to memory*) koje se izvršavaju brže nego „originalne" bat datoteke jer 4DOS učita celu takvu datoteku u memoriju i „odatle" je izvršava. MS-DOS to čini na drugačiji, ali sigurniji i sporiji način: on čita jednu po jednu naredbu, obraćajući se svaki put fajlu sa diska i pamteći dokle je stigao. E, tu sada dolazimo do odgovora na pitanje „Da li je sigurno?" (isto ono koje je u filmu „Maratonac" postavljano bezbroj puta). Ovde je odgovor „Nije" u situaciji kada iz takvog fajla učitavamo TSR programe. Pošto je btm-fajl učitan u memoriju, on ne dozvoljava TSR programu da se „naseli" neposredno iza mesta koje je pre učitavanja btm-fajla bilo „poslednje" zauzeto, nego se on „naseljava" iza btm fajla i posle izvršenja btm fajla tu ostaje neiskorišćena praznina, koja jasno nije ono što bismo želeli često da viđamo u svom računaru.

Dakle, to jeste mana, ali ako se o tome vodi računa i nema previše problema. No, ako vam *batch* datoteke nisu velike, nema neke značajne razlike, pogotovu ako je reč o brzoj mašini sa dobrim keš programom ili još bolje dobrim diskom. Ako imate veće *batch* datoteke, postoje i programi tipa *batch-to-com* (ili exe), koji taj problem sasvim lepo rešavaju, ali ne uvek i ne baš bez problema. Na *Sezam*-u se u direktorijumu IBMPC\UTIL može naći datoteka bat2ex15.zip. Radi veoma dobro, uz činjenicu da ima isti nedostatak kao i BTM u 4DOS-a, a to je da ne valja iz tako prevedenih BAT u COM datoteka postavljati TSR programe. Dakle, tu se „JP Software" pokazao i dalje na vrhu jer nije propustio da uradi nešto što je drugima pošlo za rukom.

Sledeći problem koji se javio jeste pitanje *alias*-a, to jest zamena za često korišćene niske naredbi koje sam, koristeći se blagodetima 4DOS-a, jednom postavio i sasvim zaboravio šta se zaista nalazi iza njih. Pokazalo se da MS-DOS 5.0 ima spreman odgovor na to u obliku datoteke *doskey*, koja može sasvim lepo da se podigne u UMB ili ostavi u konvencionalnoj memoriji i kojom može da se definiše onoliko *alias*-a koliko vam treba, naravno obezbeđujući da ste dobro proračunali koliko vam za to treba mesta. Prelazak sa *alias*-a za 4DOS na *alias*-e za MS-DOS pokazao se veoma jednostavnim: treba samo svakoj liniji *alias=niska* dodati na početku c:\...*doskey*, zameniti sve pojave znaka za razdvajanje naredbi u 4DOS-u onim koji koristi *doskey* i primetiti činjenicu

da je za alias-e kojima samo prenosili još neka parametre sa komandne linije potrebno dodati znak $ da bismo doskey-u naznačili da će takvog prenosa parametara biti. Na žalost, svaki od alias-a u DOS-u mora se postavljati komandom, to jest pozivanjem programa doskey. Sve se to postavlje u BAT datoteku, ali ima jednu veliku manu da se time svaki put poziva program doskey što traje prilikom učitavanja značajno duže. Može se to malo ubrzati prevođenjem BAT datoteka u izvršni program, ali to nije to. Dakle, tu je jedna prednost 4DOS-a.

No, budimo realni: alias-e učitavamo jednom kod podizanja sistema, a retko u toku rada kada nam je potrebna neka izmena. Jednom kada preskočite ova problema, sve ostalo je jednostavno.

Ipak mislimo da je tu 4DOS značajno bolji jer dozvoljava rekurzivno pozivanje alias-a, to jest moguće je iz jednog alias-a pozvati drugi. U MS DOS-u to meni nije uspelo. Takođe 4DOS vam dozvoljava da alias-e editujete neposredno na komandnoj liniji, da ih pamtite u datoteku, čitate iz datoteka i slično. Ono prvo nije moguće kod MS DOS-a 5.0, dok ostalo jeste, ali uz nešto malo muka. Tu je na polju alias-a dakle 4DOS u značajnoj prednosti. Ono što dobijemo pride jeste i editovanje na naredbenoj liniji: spisak prethodno zadatih naredbi, istina ne u onako lepom prozoru kao kod 4DOS-a (niko nije savršen) – ali to nam se vraća mogućnošću editovanja izabrane prethodne naredbe pre nego je izvršimo.

Razlike su polako nestajale pred očima i to pila se "vezanost" za 4DOS. Ostalo je da se proveri ono što je sve nas prvenstveno privuklo 4DOS-u, a to je manji otisak u konvencionalnoj memoriji u odnosu na MS DOS-ov COMMAND COM. Sa instaliranim samo himem sys-om i naredbom u CONFIG SYS u dos=high pobedio je MS DOS!! Istina, bilo je to se samo nekoliko KB, ali i to je bilo dovoljno. Odluka je pala. Neko će upitati: a zašto samo HIMEM SYS, a ne i EMM386 EXE ili još bolje QEMM386 ili 386MAX? Odgovor je jednostavan: koristim Windows-e kojima takve vrsta pomoći zaista ne treba, jer ona ništa značajno ne donosi, oduzima deo memorije, a i usporava celu mašinu zbog potrebe da se sve propušta kroz još jedno "sito", što kada se povuče crta, daje "skuplju daru nego meru". No, jasno, sve je relativno.

Objasnimo još zašto sam tako lako "pokleknuo". Ja koristim Windows-e i sve do pojave verzije 3.1 nije uvek bilo lako boriti se sa Windows-ima i 4DOS-om u trenucima kada je trebalo smirivati neke treće pakete. U slučaju COMMAND COM-a ti problemi su bili daleko manji, jer 4DOS pokušava da bi radio ono što more da prihvati dosta vezanih prekida i tu se stvari ponekad (ali istina veoma retko) komplikuju. Dalje, COMMAND COM ima brži swapping od onoga kod 4DOS-a jer je COMMAND COM daleko bliži IO SYS-u i MSDOS SYS-u. Problemi sa usporavanjem kod swapping-a se mogu izbeći njegovim isključivanjem u batch datotekama u trenucima kada to nije neophodno, ali to je već druga priča. Recimo kako 4DOS obavlja svoje swappovanje: on napravi swap file, obavi željenu operaciju i onda uporedi trenutno stanje naznačene oblasti memorije sa onim koje je "zapisao" i ako su iste sve se nastavi bez izmena, ali ako nisu onda prebaci "zapisano" stanje i opet nastavi. Za taj proces potrebno je vreme koje se zaista oseti i koje su svi korisnici doživljavali kao usporenje pri radu sa 4DOS-om u odnosu na COMMAND COM iz verzija pre 5.0. Prvo što sam želeo da probam jeste da li će se ta razlika pojaviti i kod COMMAND COM-a iz verzije 5.0 koji je otišao u HMA. Razlika je ostala, a dobit u odnosu na 4DOS je nestala. Eto objašnjenja zašto sam tako lako napustio 4DOS.

ZA ČIM ĆEMO ŽALITI

Ostaje nam da podvučemo crtu i da kažemo šta nam sada bez 4DOS-a nedostaje iz njega i može li se to prevazići. Prvo što nedostaje je prekrasna zbirka okolinskih promenljivih i funkcija nad njima koja nudi 4DOS, ali to i nije neki problem jer se sve to može iskoristiti jednostavnim pozivanjem 4dos.exe (jeste, preimenujte ga u 4dos.exe jer prezime COM i nije baš po propisu zbog njegove veličine, bilo je o tome diskusije na Sezamu, a deo toga je prenesen i u "Računarima") i obavljanjem svih željenih operacija. Dakle, ne gubimo previše. Ostaje još editovanje okolinskih promenljivih bez njihovog potpunog unošenja, to jest editovanje okolinskih promenljivih kao da je to u okviru nekog editora. Ali ne može se dobiti baš sve odjednom, zar ne? Čini se da je to mala cena. Tu je i naredba **move** za prebacivanja datoteke iz jednog poddirektorijuma u drugi, što se u MS DOS-u mora raditi uz pomoć dve naredbe ili male *batch* datoteke, ali šta je tu je. Znam da sam negde video *shareware* program MOVE i da je tu negde na nekoj od disketa, ali me mrzi da ga ponovo tražim. Čini se da nije vredno toga. Već dok bih tražio napisao bih BAT fajl koji bih preveo u COM – i eto naredbe.

Ono što mi najviše nedostaje jeste nadopunjavanje imena datoteka na komandnoj liniji i nadopunjavanje imena poddirektorijuma. Da, 4DOS verzija 4.0 je donela i novu mogućnost: umesto sa CD IME za prelazak u poddirektorijum IME dovoljno je reći IME\. Jasno nije potrebno ni celo ime, već samo početak, F9 će završiti ime ili imena ako ih više zadovoljava taj prefiks. Tu cena polako raste.

Dakle, kao zaključak recimo da je 4DOS i dalje veoma dobar proizvod ali da nije izdržao trku sa Microsoft-ovim DOS-om. Važno je napomenuti da ni Symantec nije odoleo šarmu 4DOS-a te je nekoliko nedelja pre nego je najavljena verzija 4DOS 4.0 otkupio prava prodaje (verzije 3.03) uz svoj *Norton Utilities* paket, istina sa novim imenom NDOS i sa ponešto izmenjenim stranicama. Tu je JP Software Inc napravio veoma dobar posao dobijajući veliku reklamu i verovatno dosta novca za verziju koja je bila na zalasku. Hoće li se NDOS pojaviti kao nova verzija, autoru ovoga teksta nije poznato. A u stvari to i nije važno, original je ionako uvek bolji od prepravke.

Eh da, za početnike i one koji ne vole da čitaju uputstva: 4DOS ima prekrasan help sistem koji je daleko korisniji od onoga ugrađenog u MS DOS 5.0, ali poznajući Microsoft i ovim treba biti zadovoljan. A opet 4DOS-ov help je samo 4dos.exe daleko od vas, iako koristite COMMAND COM. Dakle, sve je moguće, samo treba probati.

Možda neki od vas očekuju i konkretan odgovor na pitanje: koristiti 4DOS ili ne. Pa, ne znam. Ja sam prestao da ga koristim kao osnovnu školjku, ali svakodnevno ponešto zatreba i otkucam 4dos. Istina, sve se to desi u *Windows*-ima, pa su svi zadovoljni. To jest COMMAND COM se ne ljuti i ne oseća zapostavljenim jer je u nekom drugom prozoru. No, o *Windows*-ima možda neki drugi put ili to već bude OS/2 ili *Windows NT*, nije važno čitaoce moj. Važno je dakle reći da nikako ne valja biti odani pristalica ni jednog ni drugog. Moja preporuka je uzeti COMMAND COM kao osnovnu školjku, a 4DOS pozivati po potrebi. Ta strategija radi veoma dobro. Jasno, sve zavisi od osnovnih potreba i navika korisnika.

LOTUS 1-2-3 FOR WINDOWS

(Nastavak sa strane 23)

noj datoteci automatski znači promenu podataka u svim povezanim datotekama.

Grafička prezentacija podataka je, što se moglo i očekivati, znatno unapređena. Sasvim je u *Windows* stilu, što znači da se komandama **Graph View** ili **Graph New** podaci iz tabele prikazuju u grafičkom obliku u zasebnom prozoru. Sve osnovne promene na grafikonu se vrše iz njegovog prozora. Dodatna doterivanja grafika se rade kada se on ubaci u tabelu. Pomoću *Windows* **Draw** komandi možete napraviti zaista lep grafikon, dodavati linije, strelice ili slike iz **Clipboard**-a ako su one u *Windows* metafile ili bitmap formatima. Moguć je uvoz PIC i CGM datoteka sa diska u tabelu. Izbor tipova grafikona je dosta veliki, jer svaki tip ima po nekoliko varijanti. Najzad je 1-2-3 dobio i prave 3-D grafikone. Na raspolaganju je oko 200 različitih varijanti ovih grafikona. Rad je lagan i zabavan, jer sve se odvija kroz dijalog boksove, a miš znatno pomaže u brzini.

Štampanje tabela se odvija prema *Windows* konvencijama. Pre štampanja možete pregledati vašu tabelu (preview), izabrati printer i podesiti izgled štampane strane. Podržan je rad sa *Print Manager*-om iz *Windows*-a, tako da možete nastaviti rad dok se odvija štampanje.

Na kraju da spomenem Help. Urađen je odlično, pojavljuje se u zasebnom prozoru čiju veličinu i položaj na ekranu možete podešavati po želji. Pretraživanje je jednostavno i efikasno, a dopada mi se što se može odštampati deo helpa, tekst se može iskopirati u tabelu, a mogu se dodati i sopstveni komentari.

LEPE SLIČICE

S obzirom da sam radio sa beta verzijom, nisam hteo mnogo da kritikujem niti da hvalim, već sam samo izneo činjenice vezane za ovaj program. *1-2-3/W* je često "pucao" kroz čuvenu UAE grešku. Prilikom svakog startovanja programa to se obavezno dešavalo. Tali problem se javljao kod linkovanja datoteke i izvršavanja većih makro procedura. S druge strane, znatno je unapređen interfejs, rad sa grafikom je lagan i komotan, na raspolaganju je veliki broj tipova grafikona, a tu je svakako najveća novost za stare korisnike: prava 3-D grafika. Izuzetno zanimljiva i korisna novina su **Smart Icone** koje zaista dosta pomažu u radu čineći ga bržim i efikasnijim.

Moram ipak reći da nisam nešto oduševljen ovim programom. Gledajući sa aspekta nekog ko voli *Windows*, čini mi se da je u ovu verziju Lotus ugradio suviše kompromisnih rešenja da bi *1-2-3/W* bio bolji od *Excel*-a. Ne verujem da će iko od korisnika *Excel*-a *3.0* (verzija *4.0* se tek pojavila) preći na *1-2-3/W*, a verujte mi nema ni razloga. Ako ipak posmatram korisnika *1-2-3* koji hoće da pređe na *Windows* platformu, onda mi veoma odgovara što je zadržana stoprocentna kompatibilnost sa prethodnim verzijama, uključujući i **Clasic**. Verovatno tu "Lotus" traži svoje mesto pod suncem sa ovim programom, jer iako je ne voleli *Windows*-e, neosporna je njihova velika popularnost. Dodatni plus za *1-2-3/W* je što već sada postoje alati za kreiranje sopstvenih funkcija i makro komandi koje ćete koristiti u tabeli kao da su originalne. Reč je o "Lotus *Software Developers Kit for 1-2-3 for Windows*" u kome se kao i za prethodne verzije mogu obavljati pomenuti poslovi.

I na kraju, ako želite da se upustite u rad sa aplikacijama za *Windows*-e, onda računajte da će vam ipak trebati bar 386 računar da biste mogli da podnesete sporost u radu programa u ovom okruženju.

POVRATAK NAUCI

Da li ste ikada u otključavanju ulaznih vrata prepoznali algoritam? Ili možda u traženju određene stranice u knjizi? Algoritmi su zapravo duboko ukorenjeni u našem životu, ali o njima znamo veoma malo. Pokušajmo da serijom tekstova vratimo algoritme tamo gde pripadaju, u samo srce pravog programera.

Bojan Petrović

Algoritmi su srž i suština programiranja. Generalno, sam proces programiranja svodi se na implementiranje poznatih algoritama i eventualno pronalaženje i implementiranje novih. Šta podrazumevamo pod algoritmom? Pokušajmo da dođemo do ovog pojma posmatranjem jednog prostog i svima poznatog događaja – jutarnjeg umivanja. Naime, ako se zamislimo nad tim procesom, videćemo da ga svako jutro obavljamo na isti način, po nekom utvrđenom redosledu poznatih radnji. Svaka od tih radnji je sama za sebe opet neki skup manjih radnji, koje opet činimo po nekom redosledu. Znači, neki problem rešavamo koristeći se poznatim rešenjima koje sklapamo u kompleksniji proces pamteći samo redosled. Prema ovome algoritam ćemo shvatiti kao definisani proces ili skup pravila koji dovodi do nekog cilja.

Većina studija o algoritmima se ograničava na proučavanje elementarnih algoritama i struktura, jer je lako pokazati da se svaki problem može razložiti na gornji način, bez obzira na oblast primene. Naravno, za neke probleme možemo pronaći i više od jednog algoritma. Na primer, na koliko načina možete tražiti neku stranicu u knjizi? Moramo se, dakle, odlučiti za algoritam koji najviše odgovara datom problemu i našim mogućnostima. Lako je, naime, izabrati onaj algoritam koji oprobano daje najbolje performanse od ponuđenih, ali to ne znači da će on biti i najsrećniji izbor. Praveći neku aplikaciju, za dotični turbo algoritam ćemo utrošiti ko-zna-koliko rada na njegovoj implementaciji, još toliko na optimizaciji da bi on bio upotrebljen možda jednom i to za skup od desetak podataka. Ili, recimo, pri sortiranju niza delimično već sortiranih podataka, naš turbo algoritam će se dobro namučiti, dok će neki mnogo „lošiji" algoritam posao obaviti bez problema. S druge strane, ne možemo sebi dozvoliti ni da implementiramo pet raznih algoritama i redom ih isprobavamo, jer je to obično gubljenje vremena. Treba nam, dakle, pre nego što se pozabavimo konkretnim algoritmima, neka metoda kojom ćemo moći da ocenimo i upoznamo algoritam, a da pri tom izbegnemo sve zamke implementacije i optimizacije.

ANALIZA

Analiza algoritma se može vršiti na više načina, ali je među programerima omiljena „empirijska metoda". Naime, matematičke analize, o kojima ćemo malo kasnije, obično traže više matematičkog znanja, a posmatrane samostalno mogu da nam daju jako slabu predstavu o stvarnoj efikasnosti algoritma. Čak i kada su nam poznate matematičke studije algoritma, trebalo bi empirijski proveriti rezultate.

Empirijska metoda je ono što i inače radimo u najvećem broju slučajeva: prostim upoređivanjem vremena rada dva algoritma za očekivanu količinu podataka (i strukturu), dolazimo do nekih rezultata. Ma koliko je ovo zvučalo smešno, ovaj stari opšteprihvaćeni metod je najefikasniji način da se uverite u sposobnosti vašeg algoritma, kvalitet implementacije, ukupan efekat u vašem programu i još mnogo toga. Međutim, empirijska metoda u sebi sadrži niz zamki koje treba predvideti i uzeti u obzir. Pre svega, ova metoda se oslanja više na implementaciju nego na sam algoritam. Isti algoritam implementiran na dva načina, može imati drastično različite efekte na našu aplikaciju, resurse sistema, krajnjeg korisnika, a o brzini rada da i ne govorimo. U nekim slučajevima ćemo baš i hteti da proverimo napredak pri implementiranju jednog algoritma, kada će ova metoda biti jedini pravi način. Međutim, ako proveravamo koji je od dva ponuđena algoritma bolji za dati problem, empirijska metoda će nas posteno namučiti, pošto se teško možemo uveriti da smo obe implementacije uradili podjednako dobro (ili loše).

Lako je napraviti grešku i izborom pogrešnih ulaznih podataka. Naime, njihov broj i struktura u ogromnoj meri utiču na rad algoritma (tačnije, ta mera zavisi od tipa algoritma). Tako da treba pažljivo izabrati onaj skup koji će najvernije oslikati rad naše aplikacije. Opet, izabiranje ovog „srednjeg" skupa podataka povlači sa sobom svoje probleme, gde do izražaja dolazi i naša moć dedukcije, ali pomalo i sreća.

Konačno, jedan tehnički problem. Merenje vremena rada algoritma (što je obično i predmet empirijske analize) nije i ne treba da bude prosto očitavanje sistemskog sata na početku i kraju rada. Naime, za brze pojave je jako važna rezolucija sata koji koristimo za merenje. Recimo, algoritam kojim vršimo zapis na disk može da se poziva nekoliko stotina puta tokom rada programa, ali da svaki put sam rad traje jako kratko. Ako je rezolucija sata slaba, može se dogoditi da dobijemo da je vreme rada bilo nula! U stvarnosti, stotinak takvih „nemerljivih" operacija može da se nagomila do neočekivano dugog čekanja pred ekranom. To znači da je potrebno ili naći način da se poveća rezolucija sata, ili se setiti neke fine metode za eliminisanje tog problema. Recimo, može se koristiti petlja, u njoj naš brzi segment, a oko toga meriti vreme. Opet, evo mesta za grešku: ako petlja radi **n** obrtaja, rezultat nećemo prosto deliti sa **n**, već se prethodno moramo setiti da i sama petlja (odnosno pripadajuće instrukcije) takođe učestvuju u rezultatu. Ako ćemo da pretерujemo, i instrukcija kojom vršimo očitavanje sata i sama oduzima vreme. I tako dalje. Međutim, ako pojava koju merimo traje duže (recimo desetak sekundi) onda možda možemo i zanemariti uticaj gore pomenutih faktora.

Drugi metod analize (ali ne po važnosti) je pomenuta matematička metoda koja je pred nama. Mada matematika po definiciji predstavlja bauk za mnoge programere, ipak treba obratiti pažnju na priču koja sledi. Ove dve metode se međusobno dopunjuju i proveravaju, pa treba insistirati na korišćenju obe, jer u kombinaciji daju vrlo pouzdane i dragocene informacije. Dakle, stručno rečeno, sledi analiza složenosti algoritma.

KORAK PO KORAK

Pre svega, treba nam jedna promenljiva koja će na zadovoljavajući način opisati zadati problem. Recimo, ako posmatramo sortiranje vektora, zgodan parametar će biti dužina tog vektora. Pri radu sa matricama, red matrice je tražena promenljiva. Kod obrade stabla posmatraćemo broj njegovih čvorova ili grana. Dakle, generalno gledano, za „dimenziju problema" uzimaćemo onu veličinu koja opisuje kvantitativne osobine datog problema, a određivaćemo funkciju „složenosti" problema u zavisnosti od njegove dimenzije. Napomenimo još da se lako može dogoditi da za neki algoritam i ne pronađemo pomenutu matematičku zavisnost. Štaviše, što zbog velike složenosti, što zbog nekih nerešenih matematičkih problema, za neke algoritme nauka još nije pronašla tražene funkcije, što znači da baš vi možete biti onaj koji će računarski svet zapanjiti rešenjem nekog takvog problema.

Prvi korak u analizi algoritma je karakterizacija ulaznih podataka i odlučivanje koji metod analize treba primeniti. U nekom idealnom slučaju, za bilo koju raspodelu verovatnoća pojavljivanja nekog skupa ulaznih podataka odredili bismo odgovarajuću raspodelu vremena izvršavanja algoritma. Naravno, kako idealni slučaj nije moguć (sem kod trivijalnih problema), pokušaćemo da zadatak olakšamo: odredimo srednje vreme izvršavanja algoritma za ma koji skup ulaznih podataka, i pokažimo da će vreme izvršavanja algoritma biti manje od neke vrednosti za ma koji skup ulaznih podataka.

Drugi korak u analizi je određivanje nekih apstraktnih operacija kojima ćemo okarakterisati rad algoritma, kako bi se odvojili od same implementacije. Recimo, pri sortiranju možemo posmatrati koliko je upoređivanja potrebno za dati niz, ali i vreme potrebno da se na datoj mašini za isti niz izvrše naredbe kojima se vrši upoređivanje. Očigledno je da ovo drugo mnogo više zavisi od mašine na kojoj se vrši merenje (odnosno okoline) nego od osobina algoritma, mada oba pristupa opisuju istu stvar. Pošto je naš cilj, kao što smo ranije rekli, da izbegnemo uticaj okoline, upotrebićemo prvi pristup. Ovde zapravo spada i ideja koju ćemo koristiti: zašto se mučiti se merenjem (varljivog) vremena, kada možemo da prebrojimo instrukcije koje su se izvršile za to isto vreme?

Sledeći korak obuhvata stvarnu matematičku analizu, ali sa dva razdvojena cilja, koja smo već pomenuli: ispitivanje za srednji i za najgori slučaj. Mada na prvi pogled to ne mora tako izgledati, mnogo je lakše ispitivanje za najgori slučaj, gde zapravo želimo da dostignemo najodiji najgori slučaj za bilo koji skup ulaznih podataka. Traženje srednjeg slučaja lako može prerasti u složen matematički problem, ali njegovo rešenje će uvek isplatiti uloženi trud. U principu, srednji slučaj se svodi na izračunavanje srednjeg broja izvršavanja neke instrukcije za neku najčešću količinu i strukturu podataka, a zatim izračunavanje srednjeg vremena rada celog programa množenjem tako dobijene vrednosti za pojedinačne instrukcije sa vremenom izvršavanja instrukcije i konačno sabiranjem dobijenih vremena. Kada jednom odredimo tražene matematičke odnose, lako će nam biti da na osnovu toga odredimo, recimo, vreme izvršavanja algoritma.

Slika 1

INSTRUKCIJE	PONAVLJANJE
FOR i:=1 TO n DO	1
FOR j:=1 TO n DO	n
M[i,j]:=0	n*n
suma	n**2+n+1

```
I(n)=n**2+n+1
                 => T(n)=O(n**2)
f(n)=n**2
```

Dobijanje reda složenosti algoritama za brisanje kvadratne matrice

Slika 2

INSTRUKCIJE	PONAVLJANJE
j:=0;	1
FOR i:=1 TO n DO	1
IF niz[i]<>0 THEN	n
BEGIN	
INC(j);	n*p
niz[j]:=niz[i]	n*p
END;	
n:=j;	1
(0<=p<=1) suma	(2p+1)*n+3

```
I(n)=(2p+1)*n+3
                 => T(n)=O(n)
f(n)=n
```

Dobijanje reda složenosti algoritma za izostavljanje nula iz niza

Metod srednje vrednosti nosi nekoliko problema. Pre svega, često će biti veoma teško odrediti vreme rada date instrukcije, jer je ovakvo merenje podložno varijacama čak i na dve iste mašine za dlaku drugačije konfigurisane. Druga stvar je moguća matematička komplikacija koja se redovno pojavljuje pri traženju srednjih vrednosti, očekivanja i verovatnoće. Najgori problem je što za neke (može se pokazati da je bolje reći „mnoge") probleme nemoguće dobro definisati „srednji slučaj" (stari problem oko dobijanja niza stvarno slučajnih brojeva), pa se javlja dilema oko ispravnosti rezultata. Tako da se konačno opet vraćamo na metod najgoreg slučaja, koga je u svetlu rečenog ipak lakše obraditi.

Proces koji smo upravo opisali je ciklički, što znači da se ponavlja posle svake ispravke algoritma (ne implementacije) kako bi se realno ocenio napredak. Naravno, ceo proces se može vršiti mnogo detaljnije nego što ćemo mi to ovde činiti, ali veoma retko se pojavljuju algoritmi gde se neko veliko detaljisanje može primetno isplatiti. Proces se može završiti onda kada smo zadovoljni sa postignutim performansama ili kada više nismo u mogućnosti da zamišljena poboljšanja implementiramo.

Konačno, pre nego što krenemo sa stvarnim radom, primetimo da možemo posmatrati potrošnju više resursa računara pri ovakvoj analizi. Recimo pri radu sa diskom može nas interesovati što manji broj zapisivanja na medijum. Može nas interesovati i potrošnja memorije papira ili vremena. Kao što smo već nagovestili, mi ćemo se uglavnom ograničiti na posmatranje vremena izvršavanja algoritma, što je obično od primarnog interesa kako za programera tako i za korisnika, ali ćemo to činiti posmatranjem broja izvršenih instrukcija.

KLASIFIKACIJA ALGORITAMA

Kao što smo već rekli, za većinu algoritama možemo pronaći neki parametar koji će kvantitativno opisati problem, a najčešće je to količina ulaznih podataka. Označimo dakle ovako određenu dimenziju problema sa „n". Pošto nas zanima vreme izvršavanja, funkciju složenosti algoritama ćemo označiti sa **T(n)** (obično se naziva „vremenska složenost"). Naš cilj je da odredimo neku funkciju **f(n)** kojom ćemo zameniti teško odredivu funkciju **T(n)**, tako da ove dve funkcije za velike vrednosti **n** (što nas i interesuje) rastu istom brzinom. Takvu funkciju nazivamo „redom složenosti" algoritma, a njihovu međusobnu vezu predstavljamo sa

T(n)=O(f(n))

gde je **O** neka notacija koju, gle čuda, čitamo kao „veliko O od". Naziv funkcije **f(n)** potiče iz činjenice da za veliko **n** postoji neko **C**, takvo da je **T(n)=C·f(n)**. Naravno, u svetlu ranije rečenog, možemo (i hoćemo) posmatrati i funkciju **I(n)** koja pokazuje broj izvršenih instrukcija za datu dimenziju problema, čime se još više udaljujemo od okoline. Sada možemo izvršiti klasifikaciju algoritama prema redu složenosti, odnosno funkciji **f(n)**, pri čemu ćemo istovremeno pokazati i kako se dolazi do tražene funkcije.

f(n)=const vreme rada ovakvih algoritama je konstantno ili približno konstantno i ne zavisi od veličine problema. Ovo je tip algoritama kojima treba težiti, ali za tek poneki problem ćemo stvarno doći do ovakvog uspeha.

f(n)=log(n) vreme rada algoritma polako raste sa rastom veličine **n**.

Ovo je, imajući u vidu sporost rasta logaritma, verovatno najomiljenija vrsta problema. Ovakav tip se često pojavljuje kod programa koji svoj zadatak rešavaju transformišući ga na prostiji i time smanjujući veličinu problema na neki deo prethodnog. Klasičan postupak je polovljenja problema, gde se ukupni broj polovljenja može dobiti iz **k=log(n)**.

f(n)=n kada je vreme rada algoritma linearno, onda je obično reč o problemu gde se za svaku pojedinačnu obradu vrši mala količina rada. U ovu grupu spada najveći broj algoritama, naročito onih gde se obrađuje n istih ulaznih podataka, vrši sekvencijalna obrada i slično. Kada se n udvostruči, isto će učiniti i vreme rada. Klasičan primer je FOR petlja sa n koraka, čija se svaka obrada vrši za neko vreme t. U tom slučaju, lako se vidi da je

T(n)=n·t

f(n)=n·log(n) linearno-logaritamski problemi su obično oni koji svoj problem dele na manje delove, svaki rešavaju posebno i zatim kombinacijom takvih rešenja dolaze do konačnog. Ukratko rečeno, kada se **n** udvostruči, vreme rada se više nego udvostruči. Primer za ovakav problem je sortiranje metodom *Quicksort*, o čemu će biti reči u nekom od sledećih tekstova.

f(n)=n2** kvadratni algoritmi se najčešće dobijaju upotrebom dve ugnježđene FOR petlje, što se, recimo, radi pri obradi kvadratne matrice. Kada se **n** udvostruči, vreme rada se poveća četiri puta, pa se ovakvi algoritmi mogu praktično upotrebiti samo za manji broj problema. Na slici 1 je dat primer jednog takvog programa, zajedno sa postupkom određivanja funkcije **I(n)** koja predstavlja broj instrukcija koje se izvrše za zadato n.

Lako je razumeti da je **T(n)=t·I(n)**, gde je t prosečno vreme izvršavanja instrukcija (naravno, relacija je trebalo da bude „približno", ali smo već nagovestili da se ovde bavimo tek osnovnim razmatranjima). Pošto je odnos funkcija **T(n)** i **I(n)** dobro definisan, neće biti potrebno brinuti se o vremenu izvršavanja instrukcije, a time sa leđa skidamo teret razmišljanja o preciznom merenju vremena.

f(n)=n3** vrlo slično prethodnom, tri ugnježdene FOR petlje, tripleti elemenata i slično. Kada se n poveća dva puta, vreme izvršavanja raste osam puta. Očigledno je da će ovakvi algoritmi biti eventualno primenjivi za mali broj elemenata ili za korisnika koji ima veoma puno slobodnog vremena.

f(n)=Kn** eksponencijalni algoritmi obično pokazuju da se programer zadovoljio primenom „grube sile" pri rešavanju problema, ne razmišljajući dalje od prvog vidljivog rešenja. Naravno, ponekad drugog rešenja i nema, ali veoma retko. Očigledno je da sa n vreme rada vrtoglavo raste, tako da su ovakva rešenja praktično neupotrebljiva za iole veći broj podataka.

Ovde smo nabrojali samo neke osnovne tipove algoritama (prema redu složenosti), i to baš one sa kojima ćemo se sretati u budućim tekstovima o sortiranju, pretraživanju i ostalom. Da bi se stekao bolji utisak o stvarnim razlikama među njima, u tabeli 1 je dat uporedni (ali i aproksimativni) pregled nabrojanih tipova prema dimenziji problema.

Može se međutim dogoditi da usled složenosti problema ne možemo da dođemo do **f(n)**. Tada možemo posmatrati neku funkciju

n	log(n)	SQR(n)	n*log(n)	n**2
1 000 000	19	1000	19 000 000	1 000 000 000 000
1 00 000	16	316	1 600 000	10 000 000 000
10 000	13	100	130 000	100 000 000
1 000	9	31	9 000	1 000 000
100	6	10	600	10 000
10	3	3	30	100

Tabela uporednih vrednosti funkcije *f(n)*

g(n) čija je osnovna osobina da raste brže od **T(n)**. Naravno, uvek ćemo pronaći neku funkciju koja će trenutno da se zalepi za plafon, i time postignemo cilj. Zato naglašavamo da je trik u tome da **g(n)** raste brže ali što približnije funkciji **T(n)**. Ipak, moramo biti veoma oprezni pri tumačenju rezultata dobijenih na gornji način. Pre svega, govorimo o gornjoj granici – u stvarnosti će performanse skoro sigurno biti drugačije (verovatno bolje). Drugo, skup ulaznih podataka koji dovodi do najgoreg slučaja možda nikad nećemo sresti u praksi. Konstanta proporcionalnosti koju zanemarujemo nam zapravo nije ni poznata, pa se možemo i tu prevariti. Konkretnije, ako je vreme rada algoritma **O(f(n))**, to samo znači da programer garantuje da će vreme rada biti najviše toliko, ali verovatnije manje. Poznati algoritam *QUICKsort*, u najgorem slučaju ima vreme rada **O(n**2)**, a u stvarnosti se najčešće sreće vreme rada **O(n*log(n))**.

Na kraju, kratak zaključak. Stvarna vrednost ovakvog pristupa ocenjivanju i upoređivanju algoritama je činjenica da kompletan posao možemo obaviti na papiru, a rezultat (pogotovo ako posmatramo funkciju **f(n)**) će biti potpuno nezavisan od mašine, prevodioca i bilo čega drugog. Vreme rada se dobija množenjem gornjih funkcija sa nekim (obično) konstantnim faktorom. Uticaj tog faktora, koga mi zanemarujemo, postaje primetan tek pri većoj dimenziji problema, pa ćemo ubuduće posmatrati algoritme samo preko gornje klasifikacije. Konačno, odavde je očigledno da dva algoritma složenosti **n*log(n)** ne moraju da imaju isto vreme rada, već i veoma različito, što znači da je ovo samo jedan alat (potrebno ali ne i dovoljno merilo) u radu jednog programera.

KOJI IZABRATI

Pošto smo izvršili analize i merenja ponuđenih algoritama, ostaje nam da se na neki način odlučimo za jedan od njih. Izbor neće ili ne bi trebalo da bude prosto poređenje analizom dobijenih rezultata. Naime, sada nas čeka stvarna implementacija, koja će sa sobom doneti i uslove koji nisu bili predviđeni analizom. Na primer, relativno sporo sortiranje umetanjem (*Insertion sort*) će za delimično sortirane podatke ostaviti iza sebe neke mnogo izvikanije algoritme.

Prvi korak je i prećutno pravilo: prvo se implementira najjednostavniji algoritam koji zadovoljava naše osnovne potrebe. Ako zaključimo da rešenje i performanse zadovoljavaju, nema potrebe ići dalje. Nikada ne treba insistirati na nekom „injection-turbo-stealth" algoritmu ako stvarne potrebe za njim nema. Tako štedimo svoje vreme, a često i mnoge dragocene resurse. Ako će algoritam biti startovan samo par puta, i to još sa malim brojem ulaznih podataka, dovoljan je i prostiji algoritam. Ako će biti startovan samo par puta ali sa većom količinom podataka, tada ipak treba znati koliko to vremena može da oduzme pre nego što se odlučimo za algoritam. Važna napomena: implementacija se MORA tako izvesti da u bilo kom trenutku, ako se pokaže potrebnim, možemo izbaciti prvobitni i uvesti novi algoritam.

Ako algoritam predstavlja deo većeg sistema, tada je uobičajeno da se za početak upotrebljava pristup „grube sile". Implementiramo najprostiji algoritam na koji naletimo (ili koga se setimo) da bismo omogućili razvoj sistema. Prethodna napomena o mogućoj zameni algoritma ovde naročito mora da se uzme ozbiljno. Kako naša aplikacija bude rasla, tako ćemo se vraćati na uvedeni algoritam i razmišljati o njegovom unapređenju. Sa druge strane, dešava se i da je problem koji rešavamo takve prirode da moramo odmah pronaći najbolji algoritam. Ovakvi slučajevi se dešavaju sa aplikacijama koje, recimo, mnogo koriste neki od glavnih resursa mašine (editor sa virtuelnom memorijom na primer). Tada se kroz celu aplikaciju provlače razne zavisnosti i pozivi na kostur-algoritam, i od njegove efikasnosti i kvaliteta zavisi red celog programa. Izmena na ovakvom algoritmu povlači i menjanje većeg dela aplikacije, što će nas vrlo brzo izludeti. U ovom slučaju, pre nego što uopšte počnemo da radimo, uzimamo papir, olovku i mislimo.

Klasična greška, koju mnogi programeri nehotice prave, je da i kad se uvere da je možda potreban efikasniji algoritam, ipak ostave „lošiji" u svom delu. Naime, odluka pada na osnovu činjenice da su efikasniji algoritmi obično teži za razumevanje i implementaciju. Zato se često događa da korisnici čekaju po nekoliko puta više vremena da program završi rad, nego što bi stvarno trebalo. Kao što smo već pokazali, algoritmi složenosti **n*log(n)** su tek malo komplikovaniji od onih složenosti **n**2**, ali zato daju mnogo bolje rezultate.

Druga greška je suprotna: programer insistira na performansama algoritma. Greška se javlja onda kada se ne obrati pažnja na ulazne podatke – za njihov mali broj, algoritmi složenosti **n**2** i **n*log(n)** mogu se po efikasnosti tek malo razlikovati, a po složenosti mnogo više. Takođe, mnogi programi se pokrenu (algoritmi izvrše) tek po nekoliko puta. Vreme potrebno da se pronađe, implementira, optimizuje i proveri najefikasnije rešenje može biti daleko veće nego da smo jednostavno pustili onaj „sporiji" algoritam da obavi posao.

Prema rečenome, štos je u dobrom planiranju i tačnoj proceni. Treba balansirati između želje da po svaku cenu u naše čedo ugradimo najbolje što nauka poznaje, i želje da što lakše obavimo posao. Sa jedne strane, par dana izgubljenih na proučavanju i poboljšavanju mogu drastično pomoći krajnjem korisniku (i aplikaciji uostalom), dok sa druge strane, par dana previše možda neće dovesti ni do kakvih vidljivih efekata sem lične satisfakcije.

IMPLEMENTACIJA I OPTIMIZACIJA

Implementacija, sama po sebi, nije neki problem. Reč je o čistom programerskom zanatu, koji se svodi na primenu svojih vrhunskih znanja kako bi se zamišljeni algoritam u praksi što više približio svom pandanu u teoriji. Zahvaljujući analizama koje smo izvršili, u svakom trenutku ćemo moći da proverimo koliko je implementacija dobra i tako dovedemo svoju aplikaciju do nekog savršenstva. Ipak, analiza je samo jedna strana novčića. Sa druge strane je sistem na kom radimo, aplikacija koju radimo i njene osobine. Pri implementaciji moramo balansirati između želje da algoritam dostigne maksimum, ali i da zadovoljimo neke okvire koje nam diktira okolina. Konačno, može se dogoditi da u praksi uočimo da je algoritam koga smo analizom izabrali ipak previše zahtevan, kada se moramo vratiti par koraka unazad i pokušati sa sledećim izborom. Klasičan primer svega ovoga je sortiranje metodom *QUICKsort*, koja se može implementirati pomoću rekurzije ili bez nje. Koja rešenja ćemo koristiti, zavisiće od efekata koje želimo da postignemo, o čemu ćemo u jednom od narednih tekstova, kada će biti reči o algoritmima za sortiranje. Ili, pri stvaranju tekst editora možemo koristiti bar četiri metode čuvanja teksta u memoriji, gde se lako može pokazati da se njihove međusobne (male) razlike mogu na strani teksta meriti u desetinama kilobajta.

Optimizacija je proces stalnih pozitivnih promena koje dovode do poboljšavanja rada programa (algoritma) u bilo kom smislu. Naravno, mi ne govorimo o optimizaciji koda, kakvu rade kompajleri pri prevođenju vašeg programa, već o onim promenama koje se tiču implementacije algoritma, koju će zatim prevodioc pokušati da svari. Ovaj put moramo biti svesni da ćemo sada stvarno zavisiti od okoline, hteli mi to ili ne.

Optimizacija ima smisla samo onda kada će algoritam raditi više od nekoliko puta i to za veću količinu podataka, jer će se tek tada stvarno primetiti blagodeti optimizacije. Prvi korak je implementacija izabranog algoritma po metodi „štap i kanap", odnosno onako kako možemo u datom trenutku. Kraće rečeno, najjednostavnije ali da radi. Ovo će biti osnova za poboljšanja, a često se ispostavi da nisu ni potrebna. Sada se isprobava ranije obavljena analiza. Ako smo saznali da je algoritam složenosti **log(n)** a vreme rada se otegne, znači da smo pogrešili ili u analizi, ili u implementaciji (možda i oboje?).

Sledeći korak je određivanje „unutrašnje petlje". Naime, obično u algoritmu može da se pronađe neka instrukcija ili skup njih, koje se izvršavaju uzastopno tokom rada. Ako želimo da stvarno ubrzamo rad, prvo treba pogledati baš tu. Naime, svako rasipanje vremena u ovoj petlji se množi sa brojem njenih izvršavanja, što ume da se nagomila. Optimizacija se svodi na pažljivo pregledanje unutrašnje petlje i razmišljanje o mogućim ubrzanjima. Sada se više ne razmišlja o eleganciji, lepoti ili razumljivosti tog parčeta koda. Ako su unutra neki pozivi procedura, izbacite ih. Ako možete da ubacite malo asemblera u petlju, samo ćete još ubrzati rad. Zapravo, asembler je poslednja nada za ubrzanje, i treba joj pribeći onda kada su vam delići sekunda od životne važnosti.

Sva poboljšanja kojih se setite uopšte ne moraju da rade onako kako ste ih zamislili. Recimo, kod pomenutog *QUICKsort* algoritma izbacivanje rekurzije pomoću PUSH i POP će možda na papiru i nekim mašinama (bolje rečeno prevodiocima) stvarno ubrzati rad, ali negde drugde će se desiti suprotno (primetimo, u unutrašnju petlju ćemo ubaciti još instrukcija, i time je povećati). Ali sličan postupak implementacijom steka korišćenjem nizova i pokazivača može dovesti do višestrukog ubrzanja u odnosu na rekurzivnu verziju. Poenta je da neko veoma prosto rešenje može iznenaditi brzinom!

I da opet naglasimo, optimizacija je jako osetljiva na okolinu. Kada jednom vaš algoritam dovedete do savršenstva, ne samo da će za nekog sa strane možda biti teško razumljiv, već je moguće da na drugom prevodiocu vaši „štosovi" neće ni moći da se izvedu. Ili još gore, možda će moći da se izvedu, ali će vas rezultat potpuno obeshrabriti.

U SLEDEĆEM KROJU...

Pojam složenosti algoritma je tek jedan od pojmova sa kojima ćemo se sretati u seriji tekstova pred nama. Naime, pri obradi svakog algoritma naglasićemo kom tipu pripada (što čitalac može i sam da proveri pomoću olovke i papira), a na osnovu toga ćemo razmatrati primenljivost datog algoritma u određenim slučajevima. Tako ćemo, recimo, pokazati da i „najbrži" algoritam ponekad mora da ustukne pred najsporijim, mada to zdrava logika obično ne pogodi iz prve.

U sledećem tekstu upoznaćemo se sa nekim osnovnim strukturama koje ćemo koristiti pri obradi i implementaciji raznih algoritama, od nizova i listi, preko stekova i redova, do stabala i – rekurzija.

INDEKSIRANJE BAZE

Žalosno ali istinito, vaši su roditelji bili u pravu kada su govorili da put kroz život vodi preko indeksa. Na sreću, ovaj put se ne radi o maloj crvenoj knjižici.

Dejan Jelović

Pretraživanje baze podataka oduvek je predstavljalo vremenski zahtevan posao. Potrebno je učitati svaki, potencijalno veliki slog neke baze i uporediti ga sa onim što tražimo. Prvo rešenje ovog problema sastojalo se u tome da se u posebnu datoteku izdvoje iz svakog sloga polje po kome se vrši pretraga (takozvani ključ) i redni broj tog sloga. Sada je umesto pretraživanja gigantske baze dovoljno ključ potražiti u maloj datoteci, i iz nje uzeti redni broj sloga. Analogija sa knjigom se sama nameće: Kada vam u knjizi treba neko poglavlje potražićete ga u sadržaju, pročitati broj strane i otvoriti knjigu na pravom mestu. U skladu sa svojim papirnatim pandanom ova mala datoteka naziva se indeks.

BINARNO PRETRAŽIVANJE

Indeksi su se pokazali dobri u još jednom pogledu. Prilikom sortiranja baze po ključu mnogo je brže sortirati indeks, a bazu ostaviti onakvom kakva jeste. Druga prednost ovakvog pristupa je što je sada moguće imati više indeksa za jednu bazu. Korisnik koji želi da pregleda, recimo, službenike nekog preduzeća sortirane po imenu, a onda po radnom stažu, ne mora da čeka da program pretumba celu bazu podataka, dovoljno je samo promeniti indeks koji se koristi.

Sortirani indeksi se takođe pretražuju fantastično brže od nesortiranih. Uzmemo ključ koji se nalazi tačno na polovini sortiranog indeksa. Ukoliko tražimo vrednost veću od njega, znači da je treba potražiti u gornjoj polovini indeksa a ukoliko tražimo manju, u donjoj. Na taj način smo praktično jednim ispitivanjem prepolovili područje pretrage. Sada iz te polovine uzmemo vrednost na sredini i ponavljamo postupak sve dok ne nađemo traženi ključ. Ovakav način pretrage naziva se binarno pretraživanje. Malo matematike pokazuje da je za indeks sa **n** članova vreme običnog linearnog traženja srazmerno sa **n**, dok je dužina binarnog traženja srazmerna sa **ln n**. Kako funkcija **ln n** raste »mnogo« sporije od **n**, onda za veće baze binarna pretraga daje daleko bolje rezultate.

Nažalost, i ovde postoji „kvaka 22". Ubacivanje nove vrednosti u nesortirani indeks vrši se njenim jednostavnim dodavanjem na kraj indeksa. Kod sortiranog indeksa da bismo održali njegov raspored, moramo prvo da nađemo mesto gde naš ključ treba da stoji, onda moramo da pomerimo sve članove indeksa kako bismo napravili mesta, i tek onda upisujemo novi član. Nepotrebno je reći: pomeranje svih članova indeksa za jedan svaki put kada se ubacuje novi ključ i odgovarajuća vrednost je veoma spora operacija.

Dakle, izbor vam baš i nije sjajan. Možete da koristite nesortirane indekse u koje je lako dodavati nove članove, ali je vreme pretrage veoma sporo, ili brze sortirane indekse kod kojih svako dodavanje sloga predstavlja muku. O mon Dieu, kakve dileme! Naravno, ljudi su se snalazili krpeći algoritme za sortirane indekse tako da su ovi pružali kako-tako odbre dobre rezultate, ali takve stvari su uvek padale u domen programerskih prljavština i bile su daleko od originalne, elegantne ideje.

B-STABLA

B-stabla su 1970-te godine izmislili McCreight i Beyer. Pošto sam vas praktično vodio kroz istoriju indeksiranja, dopustite da kažem da do danas nije izmišljen bolji algoritam za ovaj posao. Svi veliki programi za baze podataka ga koriste i on predstavlja de facto standard u toj oblasti.

Ideja algoritma jeste da se od sortiranog indeksa napravi stablo, tako što će se levo od svakog čvora nalaziti čvorovi sa ključevima manjim od njega, a sa desne čvorovi sa ključevima većim od njega. Da vas ne bih dalje upetljavao u opisivanje očiglednog, pogledajte sliku sortiranog indeksa i odgovarajuće stablo. Za koren stabla uzet je broj 8, jer se on nalazio na sredini indeksa. Za čvor na koga pokazuje njegov levi vektor, tzv. levo dete, uzet je broj 4, jer se on nalazio na sredini donje polovine indeksa, a desno dete ima vrednost 13, sredinu gornje polovine indeksa, itd. Poslednji nivo stabla predstavljaju listovi, tj. čvorovi koji nemaju decu. Vektori u listovima, umesto da pokazuju na decu, sadrže vrednosti dodeljene svakom ključu indeksa (na našem primeru to su redni brojevi slogova u bazi podataka).

Pretraga B-stabla vrši se na način sličan binarnom pretraživanju. Krećući od korena upoređujemo ključeve i u slučaju da je tražena vrednost ključa manja prelazimo na levo dete, a ukoliko je veća na desno dete. Pretraga je završena ili kada nađemo traženi ključ, ili kada odlazeći do listova utvrdimo da tražene vrednosti nema u stablu. Ukoliko smo našli traženi ključ, onda treba na osnovu njega naći odgovarajuću vrednost. To radimo tako što sa čvora sa našim ključem pređemo na desno dete, a onda pratimo leve vektore sve dok ne dođemo do nekog lista. Levi vektor tog lista predstavlja traženu vrednost.

B-stabla omogućuju i prelazak sa nađenog ključa na ključeve sa od njega neposredno većim ili manjim vrednostima, tj. omogućuje sekvencijalno kretanje kroz indeks. Pri traženju, recimo, prve veće vrednosti, treba samo napraviti korak na desno dete i onda prelaziti, dokle god možemo, na levo dete. Ključ sadržan u tom čvoru je prvi neposredno veći od našeg. Ukoliko naš čvor nije imao desno dete, onda je trebalo ići uz stablo, sve dok se ne nađe čvor sa desnim detetom i tamo ponoviti ceo postupak.

OPTEREĆENI OPERATORI

Već nekoliko dana pokušavam da nađem način da spomenem polimorfne operatore u ovom tekstu. Na žalost, teško je naći vezu između operatora i baza podataka. Razlog za to je što želim da saopštim svetu da sam našao adekvatan prevod za „operator overloading". Prelistavajući prevod **Stroustrup**-ove knjige o C++-u u izdanju kod JDP-a, video sam da se koristi termin *preklapanje operatora*. Razni opterećeni, overlodirani i polimorfni operatori mogu sada da odu na zaslužen odmor. Podignite spomenik prevodiocu!

Stablo znači omogućuje pronalaženje odgovarajućeg ključa i onda sekvencijalno kretanje kroz indeks na neposredno veće ili manje vrednosti. Ta tehnika se naziva *Indexed Sequential Access Method*, ili skraćeno ISAM, fraza koju ste možda čuli i ranije.

Binarna stabla se, dakle, pretražuju na manje-više isti način kao i sortirani indeksi. Njihova prednost je u tome što omogućuju lako dodavanje novih članova. No, o tom po tom, pogledajmo prvo kako sve to izgleda.

U PRAKSI

Nadam se da vas prethodnih nekoliko pasusa nisu uplašili, jer ćemo ovde još malo da zakomplikujemo stvar. Prva komplikacija je to što svaki čvor može imati više ključeva. To je urađeno iz prostog razloga što se za čitanje čvora sa diska uvek mora učitati ceo blok, pa se veća efikasnost postiže tako što se broj ključeva tempira tako da veličina čvora odgovara veličini bloka. Sada svaki čvor može imati **k** ključeva i **k+1** vektora na ostale čvorove. Novo pravilo glasi da svaki čvor može imati maksimalno **k** ključeva, a svaki čvor osim korena mora imati minimum **k/2** ključeva. Inače princip kreiranja stabla je isti kao i ranije, samo što sada sortirani indeks od koga pravimo stablo ne delimo na polovine, nego na **k+1** delova. Pretraga stabla je izmenjena utoliko što sada nije dovoljno jedno poređenje da se utvrdi koji je čvor sledeći na redu za pretragu, nego treba pretražiti sve ključeve dok se ne utvrdi između koja dva dolazi tražena vrednost. Ukoliko čvor ima oblik

Vektor1	3	Vektor2	6	Vektor3	8	Vektor4

traženje broja 4 će ići kroz Vektor2, jer je $3 < 4 < 6$. Kako su ključevi unutar čvorova poređani po veličini, čvorovi se mogu pretraživati i binarno.

(Kada sam prvi put čuo za B-stabla bi sam ubeđen da ono „B-" u imenu stoji za „binarna". Pravilo da svaki čvor može imati više ključeva pobija ovu moju tezu, tako da je konkurs na tu temu još uvek otvoren. Nagrade slede.)

Drugi način na koji tvorci algoritma pokušavaju da nam zagorčaju život jeste pravilo da stablo mora biti izbalansirano, tj. da put od korena od svih listova mora biti isti. Iako ovaj uslov ne deluje teško kada pravimo stablo na osnovu sortiranog indeksa, ne zaboravimo da se u praksi stablo kreira u toku rada dodavanjem jednog po jednog ključa sa odgovarajućim vrednostima.

DODAVANJE KLJUČA

Dodavanje ključa počinje na isti način kao pretraga. Krećući se kroz stablo, odlazimo od najnižeg nivoa, do listova. Na tom mestu ubacujemo našu vrednost u čvor na odgovarajuće mesto. Ukoliko takav čvor sada sadrži **k** ili manje ključeva, naš posao je završen. Ukoliko smo dodavanjem napravili **k+1** ključeva, onda moramo da podelimo čvor na dva dela. Time dobijamo dva čvora sa po **k/2** ključeva, a onaj jedan viška ubacujemo u gornji, roditeljski

Slika 1

	Ime i prezime	BR
1	PERA PERIĆ	10
2	MIKA MIKIĆ	13
3	ANA KOVAČEVIĆ	2
4	BILJANA PAVLOVIĆ	4
5	DŽIPKOVIĆ SRDAN	15
6	DŽUDZEVIĆ SAMIR	8
7	NAUMOVIĆ NEBOJŠA	17
8	JELENA MILANOVIĆ	5
9	JOVAN KOČIĆ	19
10	JURA ANAK	11
11	DEJAN RAKIĆ	7
12	JELENA ILIĆ	1

BAZA PODATAKA S SLUŽBENICIMA

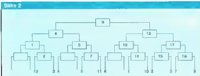
RS	BR
1	10
2	13
3	2
4	4
5	15
6	8
7	17
8	5
9	19
10	11
11	7
12	1

INDEKS BAZE PODATAKA

RS	BR
12	1
3	2
4	4
8	5
11	7
6	8
1	10
10	11
2	13
5	15
7	17
9	19

INDEKS SORTIRAN PO POLJU BR

Slika 2



B-STABLO

čvor. Ukoliko smo sada i u njemu napravili **k+1** ključ, onda ćemo i njega da podelimo na isti način i da ubacimo njegov srednji ključ u gornji čvor i tako ponavljamo proceduru sve dok eventualno ne pocepamo koren na dva dela i napravimo novi koji će da sadrži samo jedan ključ.

Stablo može ali i ne mora da dopusta držanje vise istih ključeva. Na primer, stablo identifikacionih brojeva sluzbenika ni u kom slučaju ne treba da omogući ubacivanje više istih ključeva, jer bismo onda imali vise sluzbenika sa istim identifikacionim brojem, koji naravno treba da bude jedinstven. Na drugu stranu, stablo plata svih sluzbenika treba da omogući više istih ključeva, jer se može desili da više ljudi ima istu platu.

Ukoliko u stablo ubacujemo vrednost koja već postoji, postupak je nešto komplikovaniji. Naravno, prvo se krećemo kroz stablo tražeći ključ koji treba da ubacimo. Ukoliko ga nađemo, a stablo ne dopusta više istih vrednosti, rutina za ubacivanje treba da vrati kod greške. U suprotnom, postupak za ubacivanja je sledeći. Kada smo našli ključ, treba otići do lista sa odgovarajućom vrednošću. To radimo tako što se pomerimo na čvor na koji pokazuje vektor desno od nađenog ključa, a onda od tog čvora do listova pratimo leve vektore svakog čvora. Došavši do lista, ubacujemo ključ u taj list gore opisanim postupkom.

BRISANJE KLJUČA

Ovo je jedina rutina koja je malo komplikovanija. Da bismo obrisali ključ, prvo već opisanim postupkom sićemo do odgovarajućeg lista i tu iz njega izbacimo odgovarajuću vrednost. Ako je naš ključ na nekom višem nivou, onda idemo uz stablo i iz čvorova izbacujemo leva vektore. Problem je ovde što stablo mora biti

Listing 1 BTREE.HPP

```
#ifndef __BTREE
#define __BTREE

#define ERROR  1
#define OK 0

typedef long RECNO,      // Adresa sloga ili cvora
#define NODESIZE 512     // Duzina cvora
#define MXKEYLEN 80              // Maksimalna duzina kljuca
#define ADR sizeof(RECNO)

// Greske
enum dberrors {
   I_HEAP,       // Nema mesta na HEAP u
   I_IO                  // I/O error
},

// Definisije cvora
struct NODE {
   int nonleaf,                    // Da li je list?
   RECNO prntnode,                 // Roditelj
   RECNO lfsib,                    // Levi naslednik
   RECNO rtsib,                    // Desni naslednik
   int keyct,                      // Broj kljuceva
   RECNO key0,                     // Koliko manjih kljuceva
   char keyspace [NODESIZE   ((sizeof(int) * 2) + (ADR * 4))],
   char spil [MXKEYLEN]            // Za visak pri ubacivanju
},

// Zaglavlje fajla koji sadrzi index
struct HEADER {
   RECNO rootnode,                          // Adresa korena
   int keylength             // Duzina kljuca
   int m,                    // Kljuceva po cvoru
   RECNO rlsed_node,         // Sledeci otpusten cvor
   RECNO endnode,            // Sledeci nedodeljen cvor
   int locked,               // Da li je stablo zakljucano
   RECNO leftmost            // Najlevlji  | cvor
   RECNO rightmost,          // Najdesni
}


int build_tree (char *name  int len),
```

```
class btree {
         private
         HEADER bheader,
         NODE trnode,
         RECNO currnode,          // Trenutni cvor
         int currkey              // Trenutni kljuc

         public
         FILE *fp,
         btree (void),
         btree (char *ndx_name),
         ~btree (void),
         int open (char *ndx_name),
         void close (void),
         RECNO find (void *k),
         int deletekey (void *x, RECNO ad),
         int insertkey (void *x, RECNO ad, int unique),
         RECNO nextkey (void),
         RECNO prevkey (void),
         RECNO firstkey (void),
         RECNO lastkey (void),
         void keyval (void *ky);
         RECNO currkey (void),

         protected
         virtual int compare_keys(void *a, void *b),
         virtual void dberror(void)
         virtual void read_node (RECNO nd, void *dt),
         virtual void write node (RECNO nd, void *dt),
         virtual void bseek (RECNO nd),

         private
         int treescan (RECNO *t, void *k, char **a),
         int nodescan (void *keyvalue  char **nodeadr),
         RECNO fileaddr (RECNO t, char *a),
         RECNO leaflevel (RECNO *t  char **a, int *p),
         void implode (NODE *left  NODE *right)
         void redist (NODE *left, NODE *right)
         void adopt (void *ad, int kct  RECNO newp),
         RECNO nextnode (void),
         RECNO scannext (RECNO *p, char **a),
         RECNO scanprev (RECNO *p, char **a),
         char *childptr (RECNO left, RECNO parent, NODE *btp),
         void memerr (void),
};

#endif
```

Zaglavlja klase BTREE CPP

Listing 2

```
int btree::open (char *ndx_name)

// Otvori index

{
        if ((fp = fopen(ndx_name, "rb+")) == NULL)
                return ERROR;
        fread(&dheader, sizeof(HEADER), 1, fp);
        //              Ako je stablo zakljucano, javi gresku
        if (dheader.locked)     {
                fclose(fp);
                fp = NULL;
                return ERROR;
        }
        //              Zakljucaj stablo
        dheader.locked = TRUE;
        fseek(fp, 0L, SEEK_SET);
        fwrite(&dheader, sizeof(HEADER), 1, fp);
        currnode = 0;
        currkno = 0;
        return OK;
}
```

Otvaranje datoteke sa stablom

Listing 3

```
        RECNO btree::find (void *k)

        // Nadji kljuc u stablu

    {
        int i, fnd = FALSE;
        RECNO t, ad;
        char *a;

        t = dheader.rootnode;
        if (t)    {
            read_node(t, &trnode);
            fnd = btreescan(&t, k, &a);
                ad = leaflevel(&t, &a, &i);
                if (i == trnode.keyct + 1)    {
                        i = 0;
                        t = trnode.rtsib;
                }
                currnode = t;
                currkno = i;
        }
        return fnd ? ad : ERROR;
    }

    int btree::btreescan(RECNO *t, void *k, char **a)

    // Pretrazi stablo
    {
        int nl;
        do    {
            if (nodescan(k, a))    {
                while (compare_keys(*a, k) == FALSE)
                    if (scanprev(t, a) == 0)
                        break;
                if (compare_keys(*a, k))
                    scannext(t, a);
                return TRUE;
            }
            nl = trnode.nonleaf;
            if (nl)    {
                                *t = *((RECNO *) (*a - ADR));
                read_node(*t, &trnode);
            }
        }    while (nl);
        return FALSE;
    }

    int btree::nodescan(void *keyvalue, char **nodeadr)

    // Pretrazi cvor
    {
        int i;
        int result;

        *nodeadr = trnode.keyspace;
        for (i = 0; i < trnode.keyct; i++)    {
            result = compare_keys(keyvalue, *nodeadr);
            if (result == FALSE)
                return TRUE;
            if (result < 0)
                return FALSE;
            *nodeadr += ENTLN;
        }
        return FALSE;
    }
```

Rutine za trazenje vrednosti u stablu

Listing 4

```
        int btree::compare_keys(void *a, void *b)

        // Uporedi dva kljuca

        {
            return strncmp ((char*)a, (char*)b, KLEN);
        }
```

Rutina za poređenje dva ključa

Listing 5

```
    RECNO btree::leaflevel(RECNO *t, char **a, int *p)

    // Odlazak do lišćeva na osnovu kljuca

    {
        if (trnode.nonleaf == FALSE)    { // List?
                *p = (*a - trnode.keyspace) / ENTLN + 1;
                return *((RECNO *) (*a + KLEN));
        }
        *p = 0;
        *t = *((RECNO *) (*a + KLEN));
        read_node(*t, &trnode);
        *a = trnode.keyspace;
        while (trnode.nonleaf)    {
            *t = trnode.key0;
            read_node(*t, &trnode);
        }
        return trnode.key0;
    }
```

Rutina koja na osnovu ključa traži odgovarajući list

izbalansirano, tako da procedura za brisanje proverava čvorove levo i desno od našeg (ne decu, već čvorove na istom nivou), i da ako je moguće, razdeli ključeve među njima. Takođe, ukoliko se kao rezultat brisanja dva čvora mogu skupiti u jedan (tzv. implozija), algoritam mora i to da uradi.

KLASNA PODELA

B-stablo koje ću vam ovde predstaviti vuče svoje korene iz K&R C implementacije nađene u nekoj knjizi pre sedam-osam godina. Tokom vremena je pretrpelo velike izmene, mahom radi poboljšanja efikasnosti. Zadnji potez koji sam napravio bio je da ga prelaskom na objektno programiranje zapakujem u C++ klasu, praveći tako od njega predmet obožavanja (object of beauty).

Datoteka BTREE.HPP sadrži definiciju funkcije **build_tree** i klase **btree**. Funkcija

int build_tree (char *name, int len);

pravi u datoteci name prazno stablo sa dužinom ključa **len**. Ova funkcija upisuje u datoteku neku vrstu zaglavlje koje sadrži informacije tipa gde se u datoteci nalazi koren itd. Jedna od važnijih informacija jeste i to da li je stablo zaključano. Otvaranjem datoteke sa stablom za korišćenje postavlja se indikator koja označava da je stablo zaključano, a zatvaranjem se ovaj indikator vraća na nulu. Kako stablo za sve vreme rada neke informacije drži u memoriji, nestankom struje moglo bi da dođe do korupcije podataka. U tom slučaju program neće dozvoliti korišćenje potencijalno lošeg stabla, nego treba da na osnovu podataka iz baze sagradi novo stablo.

Klasa **btree** sadrži sledeće članove:

- **int open (char *ndx_name);**

Ova rutina otvara za rad indeks (stablo) pod imenom **ndx_name**. Naravno, indeks treba da bude napravljen funkcijom **build_tree**. U slučaju greške, funkcija vraća ERROR, a u suprotnom vraća OK.

int insertkey (void *x, RECNO ad, int unique);

Ubacuje ključ i određenu vrednost u indeks. Na primer, ako smo slog radnik

struct radnikslog {
char ime [30];
int id_broj;
} radnik;

dodali u bazu podataka na mesto **brsloga**

long brsloga = database.add (&radnik);

onda bi **idbroj** tog radnika dodali u stablo naredbom

indeks.insertkey (&radnik.id_broj, brsloga, 1);

Bitno je primetiti da je RECNO isto što i **long**, tj. predstavlja redni broj sloga u bazi podataka.

- **RECNO find (void *k);**

Vraća broj sloga za dati ključ. Ukoliko dati ključ ne postoji u indeksu, funkcija vraća ERROR (-1). Dakle, broj sloga u za radnika sa odgovarajućim id_brojem našli bi naredbom

brsloga = indeks.find (&id_broj);

int deletekey (void *x, RECNO ad);

Briše ključ sa određenom vrednošću iz stabla

TABELE NA KLIPERU

Tabela je najprirodniji način prikazivanja mnogih informacija, naročito u relacionim bazama podataka. Jedna od večitih tema *Clipper* programera je kako najlakše realizovati tabelarni prikaz podataka?

Nenad Batoćanin

Tabelarni pregled podataka je u starijim verzijama *Clipper*-a rešavan pomoću funkcije **DBEdit**. Ova funkcija je solidno koncipirana i rešavala je većinu postavljenih standardnih zahteva. Međutim, bez obzira na veliki broj parametara na koje direktno utiče programer, ona je došla do svog maksimuma i sada je uspešno nasleđuju objekti iz **TBrowse** klase. Ne slučajno, ovde je objektno orijentisani pristup na izvanrednom primeru pokazao svoju veliku prednost nad imperativnim programiranjem.

Iako je u novoj verziji *Clipper*-a označena kao „zastarela", funkcija **DBEdit** još uvek nije izgubila na aktuelnosti. Veliki broj *Clipper* programera je ostao na verziji *S'87*, a nezanemarljiv broj i pored **TBrowse** objekata koristi funkciju **DBEdit**. Zato će u ovom tekstu biti govora i o funkciji **DBEdit** i o **TBrowse** objektima.

DBEDIT FUNKCIJA

Pisanje univerzalne procedure za tabelarni pregled tipa **DBEdit** veoma je interesantan (i dosta težak) problem. U suštini, treba napraviti proceduru koja će omogućiti korisniku da slobodno razgleda podatke u datoteci, da interaktivno bira slog, polje u okviru sloga i slično.

Programeri firme *Nantucket* prvobitno su problem rešili na uobičajen način: procedura koja realizuje tabelarni prikaz svakako treba da ima kao argumente koordinate dela ekrana u kome će se ispisivati podaci i niz koji sadrži opis polja koja treba prikazivati. Takva funkcija nije naročito komplikovana i može se relativno brzo napisati: treba ispisati inicijalne podatke na ekranu, a zatim tražiti od korisnika komandu. Tada se na osnovu koda pritisnutog tastera preduzima neka akcija: pomeranje kursora, skrolovanje ekrana itd. Ako se izuzmu neki tehnički detalji, skoro jednostavan problem. A praktična korist je svakako očigledna: pregledan izbor i listanje podataka.

Prva verzija funkcije **DBEdit** je imala upravo takve jednostavne mogućnosti. I novija (daleko moćnija) verzija se može koristiti na tom nivou, ponekad je i to sasvim dovoljno. Na slici 1 je prikazan primer takvog korišćenja funkcije **DBEdit**: otvori se datoteka i pozove funkcija. Treba samo zadati koordinate gde će se ispisivati podaci i navesti koja polja treba ispisati. Zaista, teško je odoleti ovakvoj jednostavnosti. **DBEdit** se može još jednostavnije pozivati: ako napišete poziv bez ikakvih argumenata, ispisuju se sva polja iz aktivne DBF datoteke preko celog ekrana.

Međutim, priča se tu nikako ne završava. Pre ili kasnije ćete doći do nekih ograničenja koja će vam smetati: prezime i ime treba pisati velikim slovima, pritisnuta strelica na dole pomera kursor na dole, osim kad podaci nisu dobro uneti, pritisak na taster <Del> bi trebao da briše tekući slog, itd. Ovakav spisak sasvim sigurno nećete nikad završiti, čak i da ga samo vi popunjavate, a kamoli korisnici prepuni ideja. Jedino rešenje ovakvih problema je dodavanje novih argumenata. Ovim se nepovratno gubi na eleganciji, ali mora se platiti cena za nove mogućnosti. Srećom, *Clipper* dozvoljava pravljenje funkcija sa promenljivim brojem argumenata, čime se omogućuje da prethodne verzije programa budu kompatibilne sa novim. Tako je funkcija **DBEdit** dobila još argumenata: nizove za definisanje zaglavlja, podnožja, separatora između kolona, stringove za formatiranje kolona itd.

Najznačajniji dodatak je funkcija koju definiše programer i koja se izvršava uvek kad korisnik pritisne neki taster. Na taj način se omogućuje da korisnik isprogramira neku svoju akciju i tako daleko poboljša fleksibilnost funkcije **DBEdit**.

Na slici 2 je prikazan jedan kompletan primer korišćenja funkcije **DBedit** sa funkcijom koju definiše korisnik. Ovaj primer možete iskoristiti kao obrazac po kome ćete kreirati tabelarni pregled u programu. U datom primeru se vrši tabelarni pregled datoteke **Adresar**, koja ima tri polja: **Ime**, **Prezime** i **Adresa**. Funkcija **DBEDit** ima 12 parametara, koje ne morate sve navoditi pri pozivu: ukoliko želite, neki parametar možete i izostaviti, s tim što će se umesto njega upotrebiti podrazumevane vrednosti (primer 1).

Kao što je rečeno, prva četiri parametra su koordinate pravougaonika na ekranu u kome će se ispisivati podaci. Sledeći parametar je niz stringova koji sadrži **izraze** koji će se prikazivati kao kolone u prikazu. Obično su to nazivi polja tekuće datoteke, ali u opštem slučaju ne mora biti tako. Na primer, umesto **"Adresa"** može stajati i **"Stranke->Adresa"** ili **"Left(GetAdres(),20)"** – znači proizvoljan izraz. Pri zadavanju izraza treba imati u vidu da **DBEdit** služi za listanje **tekuće** datoteke, tako da pomeranje kursora gore i dole pomera pokazivač sloga u aktivnoj radnoj oblasti. Ako je više datoteka povezano naredbom **SET RELATION**, bez problema se mogu prikazivati i polja iz drugih datoteka, jer tada pomeranje pokazivača sloga u tekućoj datoteci pomera i sve pokazivače u „povezanim" datotekama. Ukoliko ne navedete ovaj parametar, podrazumevaju se sva polja datoteke iz aktivne radne oblasti.

Sledeći parametar je **naziv** funkcije koju definiše korisnik i koja se izvršava posle svake operacije na prikazu. Ne treba navoditi argumente funkcije, već samo naziv.

Zatim se navodi niz stringova za formatiranje kolona: pri ispisivanju svake kolone prvo se vrši funkcija **Transform** koja menja ispis. Na primer, ako treba ispisati broj 1234, a string za formatiranje te kolone je „9,999", biće ispisano 1,234. Ovde postoje dve mogućnosti: možete navesti niz, pri čemu za svaku kolonu treba navesti po jedan format string, a možete navesti i samo jedan string, pri čemu se taj string koristi za sve kolone.

Iznad svake kolone biće natpis koji se zadaje sledećim argumentom. I ovde postoji mogućnost zadavanja niza ili samo jedne vrednosti.

Sledeća tri parametra služe za zadavanje simbola koji odvajaju zaglavlje od prvog reda prikaza, simbola koji odvajaju kolone i podnožje prikaza od poslednjeg vidljivog reda. I ovde se slično može zadati niz ili samo po jedan string koji onda važi za sve kolone.

Poslednji parametar je niz stringova koji će biti ispisani u poslednjem redu prikaza.

DBEdit funkcija radi na dva načina, u zavisnosti da li je navedena korisnička funkcija. Kada je potrebno izvršiti jednostavni pregled podataka, ne navodi se korisnička funkcija i **DBEdit** dozvoljava izbor sloga komandama za pomeranje kursora <Up>, <Down>, <Left>, <Right> itd. Pritiskom na <Enter> ili <Esc> završava se biranje i **DBEdit** završava rad. Pokazivač sloga datoteka ostaje na izabranom slogu i može se vršiti dalja obrada.

Kada se navede naziv korisnikove funkcije, **DBEdit** menja način rada. Pri svakoj akciji korisnika (pritisak na neki taster) poziva se navedena korisnikova funkcija i automatski joj se prosleđuju dva argumenta: **nMode** i **nIndex**. **nMode** je broj koji označava u kom je „stanju" trenutno **DBEdit**, na osnovu čega možemo preduzimati odgovarajuću akciju. Drugi parametar je redni broj kolone u kojoj se trenutno nalazi kursor. Korišćenjem ova dva parametra uvek možemo odrediti „šta" funkcija **DBEdit** trenutno radi.

Zbog lakšeg korišćenja, brojne vrednosti parametra **nMode** definisane su kao simboličke konstante u heder datoteci **DBEDIT.CH**. Korisnikova funkcija analizira tekuće stanje funkcije **DBEdit** i na osnovu toga preduzima određenu akciju. Moguće su sledeće vrednosti:

DE_IDLE – Korisnik je pomerio kursor nekom komandom. Funkcija data kao primer tada ispisuje tekuću poziciju kursora.

DE_HITTOP – Korisnik je upravo pokušao da pomeri kursor pre prvog sloga datoteke. Funkcija **DBEditUDF** ispisuje poruku da je korisnik došao do početka datoteke.

DE_HITBOTTOM – Korisnik je pokušao da pomeri kursor iza poslednjeg sloga u datoteci. Ispisuje se odgovarajuća poruka.

DE_EMPTY – Nema slogova u tekućoj radnoj oblasti. Funkcija ispisuje odgovarajuću poruku.

DE_EXCEPT – Korisnik je pritisnuo neki taster koji funkcija **DBEdit** ne prepoznaje. To može biti bilo koji taster osim onih za pomeranje kursora u **DBEdit**-u. Kada **DBEdit** pošalje ovaj kod, naša funkcija treba da preduzme odgovarajuću akciju. U svim prethodnim slučajevima naša funkcija je morala da „obavesti" na neki način funkciju **DBEdit** šta treba uraditi. Funkcija **DBEditUDF** je vraćala vrednost **DE_CONT** (takođe definisanu u **DBEDIT.CH**) koja označava da se rad normalno nastavi. U ovom slučaju možemo vraćati i neke druge vrednosti.

Na primer, korisnik je pritisnuo taster <Esc> čime označava da želi da prekine rad. Pošto naša funkcija detektuje taj pritisak, funkciji **DBEdit** se nalaže da prekine rad tako što joj šaljemo vrednost **DE_ABORT**. Ostale komande se slično dodaju. Na primer, pritisak na taster <Ins> dodaje prazan slog, a <Del> briše tekući slog. Posle ovih akcija, **DBEdit** funkciji se vraća indikator **DE_REFRESH** koji označava da treba ponovo ispisati sliku na ekranu, jer je promenjena. Sasvim slično se realizuje i izmena polja na kome je kursor.

Funkcija **DBEdit** je dosta dobro sredstvo

Silka 1

```
PROC Main

   USE Adresar NEW

   // Tabelarni pregled
   DBEdit( 0,0,24,79, { "Ime", "Prezime", "Adresa" })

   CLOSE Adresar

RETURN
```

Jednostavno korišćenje funkcije *DBEdit*

Silka 2

```
#include "DBEdit.ch"
#include "Inkey.ch"

//
PROC Main
//
   PRIVATE aFields := {"Ime", "Prezime", "Adresa"}

   USE Adresar NEW

   DBEdit( 0,0,23,79, ,                    // Koordinate
      aFields,                             // Polja
      "DBEditUDF", ,                       // Korisnicka f-ja
      "@!",                                // Formati
      {"IME", "PREZIME", "ADRESA I BROJ"}, ; // Naslovi
      "═",                                 // Ispod naslova
      " │ ",                               // Separator
      "─",                                 // Iznad podnozja
      " │ "                                // Podnozje
      )

   CLOSE Adresar

RETURN



// Kontrolna funkcija za DBEdit

//
FUNC DBEditUDF( nMode, nIndex)
//
   LOCAL nRet, Ch, nSel

   // Sta javlja DBEdit?
   DO CASE

      // Pritisnut je neki taster za pomeranje kursora,
      // nastavi sa normalnim radom
      CASE nMode == DE_IDLE
         // Pozicija kursora
         Msg( "Record " + Str(RecNo(), 3,0) + ;
                 " Column " + Str(nIndex, 3,0))
         nRet := DE_CONT

      // Korisnik pokusava da pomeri kursor pre pocetka fajla
      CASE nMode == DE_HITTOP
         Msg( "** Pocetak datoteke **")
         nRet := DE_CONT

      // Korisnik pokusava da pomeri kursor posle kraja dat
      CASE nMode == DE_HITBOTTOM
         Msg( "** Kraj datoteke **")
         nRet := DE_CONT

      // Nema podataka u datoteci
      CASE nMode == DE_EMPTY
         Msg( "** Datoteka je prazna **")
         nRet := DE_CONT
      // Korisnik je pritisnuo neki taster koji nije
      // komanda DBEdit-u
      CASE nMode == DE_EXCEPT

         // Korisnik je pritisnuo?
         Ch := LastKey()

         // Sta korisnik zeli?
         DO CASE

            // Exit
            CASE Ch == K_ESC
               nRet := DE_ABORT

            // Append blank
            CASE Ch == K_INS
               APPEND BLANK
               Msg( "** Novi slog dodat **")
               nRet := DE_REFRESH

            // Delete
            CASE Ch == K_DEL
               DELETE
               PACK
               Msg( "** Slog je izbrisan **")
               nRet := DE_REFRESH

            // Izmena sloga
            CASE IsAlpha(Chr(Ch)) .OR. IsDigit(Chr(Ch))
               KEYBOARD Chr(K_ENTER) + Chr(Ch)
               nRet := DE_CONT

            // Izmena sloga
            CASE Ch == K_ENTER
               // Naziv promenljive koju treba menjati
               cGet := aFields[nIndex]
               SetCursor(1)
               @ Row(), Col() GET &cGet
               READ
               SetCursor(0)
               nRet := DE_CONT

         END CASE

   END CASE
RETURN nRet



// Ispisivanje poruke u poslednjem redu

//
PROC Msg( cTxt)
//
   @ 24,0
   @ 24,0 SAY cTxt
RETURN
```

Primer korišćenja funkcije *DBEdit*

Silka 3

```
#include "inkey.ch"

//
PROC Main
//
   LOCAL TB, Ch
   LOCAL aFields := { "Ime", "Prezime", "Adresa" }

   CLS
   // Otvaranje datoteke
   USE Adresar NEW

   // Kreiranje TBrowse objekta
   TB := TBrowseDB( 0,0,24,79)

   // Dodavanje kolona TBrowse objektu
   TB:addColumn( TBColumnNew( "IME", { || Ime }))
   TB:addColumn( TBColumnNew( "PREZIME", { || Prezime }))
   TB:addColumn( TBColumnNew( "ADRESA", { || Adresa }))

   // Postavljanje raznih parametara
   TB:headSep := "═"
   TB:colSep := "│"
   TB:freeze := 1

   // Glavna petlja za kontrolu prikaza
   WHILE .T.

      // Vrsi se ispisivanje ekrana, sve dok se ne ispise ekran ili
      // dok korisnik ne pritisne neki taster
      Ch := 0
      WHILE !TB:stable .AND. Ch == 0
         Ch := Inkey()
         TB:stabilize()        // Ispisi jedan red ekrana
      END DO

      // Ako je ekran ispisan, a korisnik nije nista pritisnuo
      // cekaj na komandu
      IF TB:stable .AND. Ch == 0
         Ch := Inkey(0)
      END IF

      // Analiza pritisnutog tastera
      DO CASE

         // Kraj
         CASE Ch == K_ESC
            EXIT

         // Append
         CASE Ch == K_INS
            APPEND BLANK
            Fresh(TB)

         // Delete
         CASE Ch == K_DEL
            DELETE
            PACK
            Fresh(TB)
         // Izmena sloga
         CASE IsAlpha(Chr(Ch)) .OR. IsDigit(Chr(Ch))
            KEYBOARD Chr(K_ENTER) + Chr(Ch)

         // Izmena sloga
         CASE Ch == K_ENTER
            // Ispisi ekran
            Fresh(TB)

            // Naziv promenljive koju treba ucitati
            cSel := aFields[TB:colPos]
            SetCursor(1)

            @ Row(), Col() GET &cSel
            READ

            SetCursor(0)

            // Ponovno ispisivanje tekuce vrednosti
            TB:refreshCurrent()

         // Pomeranje kursora
         OTHERWISE
            MoveKey(TB, Ch)

      END CASE
   END DO

   CLOSE Adresar
```

Nastavija se na sledećoj strani!

```
RETURN


// Analiza tastera za pomeranje kursora

//---
PROC MoveKey (TB, Key)
//---
   DO CASE
      CASE Key == K_UP          ; TB:Up ()
      CASE Key == K_DOWN        ; TB:Down ()
      CASE Key == K_LEFT        ; TB:Left ()
      CASE Key == K_RIGHT       ; TB:Right ()
      CASE Key == K_HOME        ; TB:Home ()
      CASE Key == K_END         ; TB:End ()
      CASE Key == K_PGUP        ; TB:PageUp ()
      CASE Key == K_PGDN        ; TB:PageDown ()
      CASE Key == K_CTRL_PGDN   ; TB:goBottom()
      CASE Key == K_CTRL_PGUP   ; TB:goTop()
      CASE Key == K_CTRL_LEFT   ; TB:panLeft()
      CASE Key == K_CTRL_RIGHT  ; TB:panRight()
      CASE Key == K_CTRL_HOME   ; TB:panHome()
      CASE Key == K_CTRL_END    ; TB:panEnd()
   END DO
RETURN


   // Ispisivanje slike

   //---
   PROC Fresh (TB)
   //---
      LOCAL Key

      // Ako je EOF, vrati se na kraj
      IF Eof() ; GO BOTTOM ; END IF

      // Zapamti vrednost indeksa, ako je aktiviran
      Key := IF( Empty(IndexKey()), NIL, &(IndexKey()) )

      TB:refreshAll()
      WHILE !TB:stabilize() ; END DO

      IF Key != NIL
         WHILE &(IndexKey()) > Key .AND. ! TB:hitTop()
            TB:up()
            WHILE !TB:stabilize() ; END DO
         END DO
      END IF

   RETURN
```

Primer korišćenja *TBrowse* objekta

Slika 4

```
         // Kraj
         CASE Ch == K_ESC        ; EXIT
         // Pomeranje kursora
         CASE Ch == K_UP         ; TB:Up ()
         CASE Ch == K_DOWN       ; TB:Down ()
         CASE Ch == K_LEFT       ; TB:Left ()
         CASE Ch == K_RIGHT      ; TB:Right ()
         CASE Ch == K_HOME       ; TB:Home ()
         CASE Ch == K_END        ; TB:End ()
         CASE Ch == K_PGUP       ; TB:PageUp ()
         CASE Ch == K_PGDN       ; TB:PageDown ()
         CASE Ch == K_CTRL_PGDN  ; TB:goBottom()
         CASE Ch == K_CTRL_PGUP  ; TB:goTop()
         CASE Ch == K_CTRL_LEFT  ; TB:panLeft()
         CASE Ch == K_CTRL_RIGHT ; TB:panRight()
         CASE Ch == K_CTRL_HOME  ; TB:panHome()
         CASE Ch == K_CTRL_END   ; TB:panEnd()

      END CASE
   END DO

RETURN


// Skip funkcija

//
FUNC MySkip (n)
//
   LOCAL nSkip

   DO CASE
      // Ne treba pomerati pokazivac
      CASE n == 0 ; nSkip := 0

      // Na dole
      CASE n > 0
         IF i + n <= Len(aData)
            i += n
            nSkip := n
         ELSE
            nSkip := Len(aData) - i
            i := Len(aData)
         END IF

      // Na gore
      CASE n < 0
         IF i + n >= 1
            i += n
            nSkip := n
         ELSE
            nSkip := 1 - i
            i := 1
         END IF

   END CASE

RETURN nSkip
```

```
#include "inkey.ch"

//
PROC Main
//
   PUBLIC TB, Ch
   PUBLIC aData[70,10], i

   // Popunjavanje matrice vrednosti
   FOR i := 1 TO Len(aData)
      FOR j := 1 TO Len(aData[i])
         aData[i,j] := Str(i,2,0) + Str(j,2,0)
      NEXT
   NEXT

   CLS
   // Kreiranje TBrowse objekta
   TB := TBrowseNew (0,0,23,39)

   // Definisanje kolona TBrowse objekta
   FOR j := 1 TO 10
      cBody := "aData[i," + Str(j,2,0) + "]"
      TB:addColumn (TBColumnNew ("", { || &cBody }))
   NEXT

   // Postavljanje raznih parametara
   i := 1
   TB:headSep := "ÍÑÍ"
   TB:colSep := " ³ "
   TB:freeze := 1

   // Blokovi za pozicioniranje kursora
   TB:skipBlock := { |n| MySkip(n) }
   TB:goTopBlock := { || i := 1 }
   TB:goBottomBlock := { || i := Len(aData) }

   // Glavna petlja se komandama
   WHILE .T.

      // Ispisivanje ekrana
      WHILE !TB:stabilize() ; END DO

      // Status poruka
      @ 24,0
      IF TB:hitTop
         @ 24,0 SAY "** Početak niza **"
      ELSEIF TB:hitBottom
         @ 24,0 SAY "** Kraj niza **"
      ELSE
         @ 24,0 SAY "Vrsta: " + Str(i,2,0) + "  Kol:" + Str(TB:colPos,2,0)
      END IF

      // Čeka se komanda
      Ch := Inkey(0)

      // Analiza pritisnutog tastera
      DO CASE
```

Primer korišćenja *TBrowse* objekta za tabelarni pregled matrice

Slika 5

```
#include "inkey.ch"

STATIC aFiles := {}, aTB := {}, nCurrent
//
PROC Main
//
   LOCAL TB

   SET SCOREBOARD OFF
   CLS

   WHILE .T.

      // Ako nema otvorenih prozora, otvori neki
      WHILE Len(aTB) == 0 ; NewBrowse() ; END DO

      // Ispisi sliku
      WHILE !aTB[nCurrent]:stabilize() ; END DO
      Ch := Inkey(0)

      // Zapamti tekuci broj sloga
      aTB[nCurrent]:cargo := RecNo()

      DO CASE
         CASE Ch == K_ESC      ; EXIT
         CASE Ch == K_ENTER    ; NewBrowse ()

         CASE Ch == K_TAB
            IF nCurrent < Len(aTB)
               nCurrent++
            ELSE
               nCurrent := 1
            END IF
            TB := aTB[nCurrent]
            Msg ("Prozor broj " + Str(nCurrent,3,0))
            @ TB:nTop-1,TB:nLeft-1 TO ;
              TB:nBottom+1,TB:nRight+1 DOUBLE
            SELECT (aFiles[nCurrent])
            GO TB:cargo
            TB:refreshAll()

         CASE Ch == K_UP       ; aTB[nCurrent]:Up ()
         CASE Ch == K_DOWN     ; aTB[nCurrent]:Down ()
         CASE Ch == K_LEFT     ; aTB[nCurrent]:Left ()
         CASE Ch == K_RIGHT    ; aTB[nCurrent]:Right ()
         CASE Ch == K_HOME     ; aTB[nCurrent]:Home ()
         CASE Ch == K_END      ; aTB[nCurrent]:End ()
         CASE Ch == K_PGUP     ; aTB[nCurrent]:PageUp ()
         CASE Ch == K_PGDN     ; aTB[nCurrent]:PageDown ()
      END CASE
   END DO
RETURN
```

```
// Kreiranje novog objekta

//
PROC NewBrowse ()
   LOCAL cFName := Space(8)
   LOCAL x1,y1,x2,y2, cScr, Ch

   SetCursor (1)
   @ 24 0 SAY "UNESITE NAZIV NOVE DATOTEKE " GET cFName
   READ
   SetCursor (0)

   // Da li postoji datoteka?
   IF File (AllTrim(cFName) + ".DBF")

      // Odredi mesto za novi prikaz
      x1 := y1 := 0
      x2 := 10 ; y2 := 27

      Msg ("<Enter> za kraj, strelice za pomeranje  <PgDn> i <PgUp> velicina")

      WHILE .T.

         cScr := SaveScreen (x1,y1,x2,y2)
         @ x1,y1 TO x2,y2
         Ch := Inkey(0)
         RestScreen (x1,y1,x2,y2,cScr)

         DO CASE
            CASE Ch == K_ENTER
               @ x1,y1 TO x2,y2 DOUBLE
               x1++ ; x2-- ; y1++ ; y2--
               EXIT

            CASE Ch == K_UP  AND  x1 > 0
               x1-- ; x2--

            CASE Ch == K_DOWN  AND  x2 < 23
               x1++ ; x2++

            CASE Ch == K_LEFT  AND  y1 > 0
               y1-- ; y2--

            CASE Ch == K_RIGHT  AND  y2 < 79
               y1++ ; y2++

            CASE Ch == K_PGUP .AND  x2 < 23  AND  y2+3 <= 79
               x2++ , y2 += 2

            CASE Ch == K_PGDN  AND  x2 > x1 .AND  y2-3 > y1
               x2-- ; y2 -= 3

         END CASE
      END DO

      // Da li je datoteka vec u spisku?
      IF AScan (aFiles, cFName) == 0
         USE (cFName) NEW
      END IF

      // Dodavanje novog elementa
      AAdd (aFiles, cFName)
      AAdd (aTB, TBrowseDB(x1,y1,x2,y2))
      nCurrent := Len(aTB)
      aTB[nCurrent]:headSep := "-"
      SELECT (cFName)

      // Sve kolone ulaze u prikaz
      FOR i := 1 TO FCount()
         cBody := Field(i)
         aTB[nCurrent]:addColumn (TBColumnNew (Field(i), { || &cBody }))
      NEXT

      Msg ("Prozor broj " + Str(nCurrent,3,0))

   ELSE
      Msg ("Datoteka ne postoji!")
   END IF
RETURN

//
PROC Msg (cTxt)
//
   @ 24 0
   @ 24 0 SAY cTxt
RETURN
```

Primer kontrola više *TBrowse* objekata

za kreiranje tabelarnih prikaza, ali je daleko od savršenog. Korišćenjem raznih trikova mogu se postići razni interesantni efekti: da nekoliko kolona prikaza bude „zamrznulo" da se neke kolone prikazuju posebnim bojama itd, ali sve se to postiže nepreporučljivim tehnikama koje rezultuju veoma nerazumljivim i „prljavim" kodom. *Clipper 5.0* sadrži daleko moćnije sredstvo za realizaciju tabelarnih prikaza – klasu objekata **TBrowse**.

TBROWSE OBJEKTI

Funkcija **DBEdit** je u koncepcijskom smislu osuđena na neka ograničenja. Naime, ona je konstruisana da obavlja jedan složen posao i tako korisnika oslobodi zamornih tehničkih detalja, pa znači da korisnik ne treba bitno da utiče na njen rad. Ali, sa druge strane, da bi se postigla potrebna fleksibilnost, mora se omogućiti da korisnik utiče na ponašanje ove funkcije, i to na jednom vrlo osetljivom nivou. Te dve stvari su potpuno suprotstavljene i ništa ih ne može do kraja pomiriti. Čak i **TBrowse** objekti nisu konačno, već samo bolje rešenje.

Svaki **TBrowse** objekt je jedan jednostavan skup podataka i procedura (funkcija) sasvim sličan tipu **RECORD** u paskalu. Neke od internih promenljivih **TBrowse** objekta su prosta zamena za parametre koji se prosleđuju funkciji **DBEdit**. Na primer, interna promenljiva **headSep** sadrži simbole koji definišu granicu između zaglavlja tabele i podataka. Kod **DBEdit** funkcije smo ovu vrednost slali kao argument, a sada je dovoljno izmeniti internu vrednost odgovarajuće promenljive objekta

TB:headSep := "+"

Pri tome je **TB** objekt iz klase **TBrowse**. Na **slici 3** je prikazan primer korišćenja ovih objekata koji približno odgovara korišćenju funkcije **DBEdit** iz prethodnog primera.

Prvo se kreira novi objekt iz klase **TBrowse** i dodeljuje se promenljivoj **TB**. Zatim se zadaju kolone koje treba prikazivati. Kolone se zadaju korišćenjem tzv. kodnih blokova, o čemu je već bilo reči u „Računarima". Prvo se kreira objekt iz klase **TBColumn** pozivom funkcije **TBColumnNew** i time se definiše izgled jedne kolone. Zatim se taj objekt „dodaje" kreiranom **TBrowse** objektu i na taj način se specificiraju redom sve kolone prikaza. Razni parametri prikaza se menjaju izmenom internih promenljivih kreiranog objekta. U datom primeru se menjaju separatori i broj zamrznutih kolona.

Ovaj pristup se razlikuje koncepcijski od poziva funkcije **DBEdit**: umesto da se sada pozove određena funkcija koja će nešto „raditi", korisnik i dalje zadržava aktivnu kontrolu, čime je omogućena visoka fleksibilnost. Dalji rad se obavlja tako što se u ciklusu „naređuju" razne operacije **TBrowse** objektu, a stanje objekta se čita iz njegovih internih promenljivih. Objektu se naređuje tako što mu se šalju „poruke" – u stvari, startuju se interne procedure tog objekta.

Na početku ciklusa se ispisuje ekran. Objekt **TB** vodi internu evidenciju koje slogova treba ispisati (na početku su svi slogovi označeni kao neispisani). Interna funkcija **stabilize()** ispisuje prvi od neispisanih slogova. Na slici se vidi da se ta funkcija poziva u ciklusu sve dok se i poslednji markiran slog ne ispiše (tada će interna promenljiva **stable** imati vrednost .T.) ili dok korisnik ne pritisne neki taster. Tada se ispisivanje prekida i nastavlja se tek po obradi pristigle komande. Ako neki slog ostane neispisan (slika na ekranu nije potpuna), ispisaće se prilikom sledećeg dolaska na ovaj niz naredbi.

Kada se ispiše ekran, pomoću funkcije **Inkey()** se korisniku omogućuje da zada neku komandu. Ostatak programa služi za analizu pritisnutog tastera. Sasvim prirodno su tu realizovani kraj rada, izmena, dodavanja i brisanje sloga, veoma slično kao kod funkcije **DBEdit**.

Programer mora sam definisati i odgovarajuće akcije za pomeranje kursora. Srećom postoje ugrađene interne funkcije koje služe za pomeranje kursora, na vama je jedino da ih aktivirate kad korisnik pritisne određeni taster. Pošto tastera za pomeranja kursora ima prilično, zgodno je njihovu analizu izdvojiti u posebnu proceduru (**MoveKey**), koju možete koristiti i u drugim sličnim prilikama.

Posebno objašnjenje zahteva slučaj kad treba regenerisati sliku na ekranu. Već je rečeno da **TBrowse** objekt interno markira sve slogove koje treba ispisati i prilikom sledećeg poziva funkcije **stabilize()** on ih redom ispisuje. Kada se izmeni polje u nekom slogu, da bi se izmena prikazala na ekranu, potrebno je taj slog označiti kao „neispisan". To se postiže funkcijom **refreshCurrent()**. Problem je ako se na neki način izmeni cela slika, na primer, doda se novi slog, korisnik određenom komandom izvrši skok na neki slog itd. Postoji funkcija **refreshAll()** koja označava sve vidljive slogove kao neispisane, ali to nije uvek dovoljno. Naime, **TBrowse** objekt uvek teži da zadrži kursor na mestu na kom je trenutno. To u nekim situacijama može rezultovati da se slika na ekranu „zabrlja". Na primer, kad se izvrši skok na kraj datoteke, može se desiti da nema dovoljno slogova za prikaz na ekranu i tada **TBrowse** objekt pomera kursor gore dok ne uskladi sliku. Ovaj jednostavni algoritam prestaje da važi ako se menja slog u indeksiranoj datoteci – ukratko, tada **TBrowse** objekt često „zabrlja" sliku. Zato umesto funkcije **refreshAll()** treba koristiti funkciju **Fresh()** koja ispravlja ovaj nedostatak.

PREGLED MATRICA

TBrowse objekte karakteriše visoka fleksibilnost i to ne samo u pregledu DBF datoteka. Moguće je napraviti tabelarni pregled i drugih vrsta podataka. Na primer, jednostavno je napraviti tabelarni pregled nekog dvodimenzionalnog niza. Naime, interna struktura **TBrowse** objekta je relativno jednostavna. On „ne zna" (i nije mu važno) sa kakvim podacima radi – kompletna komunikacija se svodi na izvršavanje odgovarajućih funkcija za pomeranje tekuće pozicije u podacima koji se prikazuju. Na primer, kad se izvrši **TB:down()** za pomeranje

Nastavlja se na strani 60

IZ ŠUPLJEG U PRAZNO

U prošlom nastavku ove serije smo postavili temelje editora – odlučili se za sistem pamćenja jednog reda teksta, definisali odgovarajuće strukture podataka koje će sadržati neophodne informacije, deklarisali globalne promenljive i izveli dve funkcije – inicijalizaciju i funkciju za prikaz teksta na ekranu. U prizemlje ove kuće spada funkcija za unos teksta i glavna petlja programa – sve ostalo su spratovi kojih može biti koliko god želite.

Zoran Živetić

ISPIS REDA

Predstavljajući funkciju za prikaz teksta kao maksimalno zatvorenu celinu, koja treba u potpunosti da ostatak programa reši brige sa prikazom, jedan detalj je izostavljen. Radi se o funkciji **movetoscr**, koja je skrivena u dubini **display** funkcije. Pojavljuje se u sledećem kontekstu

movetoscreen(int xpos, int ypos, char *txt, int len, int framew)

i zadatak joj je da na ekran prenese jedan red teksta. Usput, u „Računarima" broj 81, u listingu **display** funkcije (kraj slike 4 na strani 59), baš u ovom pozivu **movetoscreen** potkrala se jedna greška – iz parametara **ypos** je izostavljeno sabiranje promenljive **scrline**. Umesto **tf.framey+n** treba da stoji **tf.framey+scrline+n**.

Još u prvom tekstu iz serije bilo je reči o problemima oko izbora metode za ispis na ekran. Direktno „gađanje" video memorije je dugo vremena bilo jeres za programere, ali je na kraju praksa pobedila teoriju. Ipak, može se i dalje pretpostaviti situacija u kojoj direktan upis u video memoriju zaista nije poželjan (pre svega u okruženju drugih operativnih sistema, gde često nije uopšte ni moguć), pa je zato prenos sadržaja na ekran poveren posebnoj funkciji, kako bi se izmena i prilagođenje editora okruženju moglo lako izvesti.

Najjednostavnije bi, svakako, bilo upotrebiti neku C-ovsku prenosivu funkciju poput **puts** ali su, na žalost, ovakve funkcije prespore za upotrebu u najkritičnijoj tački osnovne brzine editora. Treba imati na umu da su u nekim specifičnim slučajevima, recimo editorima koji treba da rade na nekom terminalu ili pod nekim grafičkim okruženjem, na raspolaganju samo izuzetno spore metode ispisa (zamislite, recimo, editor na nekom BBS-u, koji prima podatke brzinom od 240 znakova u sekundi!), pa nemamo mnogo izbora – treba razmotriti šta se može učiniti čak i sa ovako sporim funkcijama. Rešenje je u daljem „usitnjavanju porcija" koje treba ispisati. U našem primeru je najsitniji element jedan red – nema nikakvih prepreka da se i on rastavi na dalje tri slučaja, kada treba ispisati samo jedan znak, ceo red, ili ostatak reda – od kursora do kraja po istom principu kako smo to uradili po vertikalnoj podeli ekrana.

Ipak, za PC i DOS okruženje koje u ovom tekstu pretpostavljamo, ispis celog reda, svaki put kada se promeni makar i jedan znak u njemu, moguće je dovoljno brzo obaviti, pa nema potrebe za daljim komplikovanjima. Na slici 1 je data konkretna realizacija **movetoscr** funkcije koja koristi direktan pristup video memoriji. Treba primetiti da je u funkciju, jednostavnosti radi, ukodirana adresa video bafera, što svakako ne može ostati u tom obliku u konačnom programu. O ispitivanju tipa ugrađenog video adaptera i određivanju adrese je bilo do sada reči u „Računarima", pa se time neću baviti.

GLAVNA PETLJA

Iako bi, po nekom logičnom redosledu, pre pogleda na glavnu petlju programa trebalo prvo pokazati kako se unosi ili izbacuje znak iz reda, ipak ću, iz dva razloga, prvo pokazati kako izgleda glavna petlja. Prvi razlog je jednostavnost, a drugi činjenica da se njom, čak i bez unosa znakova, dobijamo sasvim upotrebljiv program, kojim se mogu pregledati datoteke na ekranu. Uz funkcije za ispis, pomeranje kursora, učitavanje teksta i glavnu petlju dobijamo LIST program.

Glavna petlja treba da bude zaista jednostavna. Njen jedini zadatak je da primi ulaz sa tastature i da nizom **case** situacija usmeri izvršavanje na neku od funkcija programa. Primer **main** funkcije sa nekoliko osnovnih situacija je dat na slici 2. Verujem da ne zahteva skoro nikakav komentar. Jedina specifičnost je izvršavanje **display** funkcije u svakom ciklusu petlje. Iako ovo na prvi pogled može delovati kao usko grlo, jer će se funkcija pozivati čak i ako je pritisnut neki taster koji nema nikakvo dejstvo, to ipak nije tačno. Ako se prisetite slike 4 iz prošlog nastavka, setićete se da je aktivnost unutar funkcije određena statusnom promenljivom **tf.refresh**. Ako je njeno stanje REFRESH_NONE biće izvedeno svega nekoliko instrukcija (nekih četiri poređenja **int** vrednosti i dve dodele **int** vrednosti), što praktično traje beskonačno kratko.

Da biste i bez ostatka teksta, sa ovim što je do sada rečeno, mogli da uradite nešto konkretno, na slici 3 je data jedna siromašna ali za tu priliku dovoljna verzija funkcije **loadtext**. U **main** funkciju dodajte pre same petlje instrukciju

loadtext(argv[1]);

i dopišite osnovne funkcije za pomeranje kursora sa slike 4. Time dobijate potpuno funkcionalan program LIST, kojim možete pregledati sadržaj tekstualnih datoteka na ekranu. Ako iskoristite i varijantu o kojoj je bilo reči u prošlom nastavku, i omogućite rad sa više tekstova istovremeno, verujem da ćete dobiti dosta novih ideja kako se i ovakav program može korisno upotrebiti. Jedna od njih bi, recimo, bio program za vizuelno poređenje tekstova (da li ste nekada našli na dve verzije svog programa – u jednoj od njih je učinjena neka sitna izmena, ali na kojoj i gde?). Dovoljno je da u inicijalizacionoj rutini otvorite dva radna teksta, da ekran podelite vertikalno na pola (menjajući vrednosti **framex/y/w/h** u **TEXTFILE** strukturi) i da u glavnoj petlji unesete duple pozive rutine za pomeranje kako bi pomeranje oba teksta bilo potpuno sinhronizovano.

UMETANJE...

Pošto ste zadovoljili potrebu da već sada imate nešto konkretno od ove serije, možemo nastaviti dalje. Na redu je još jedan, poslednji od kamena temeljaca – funkcija za unos znaka u red.

Ako se podsetimo na način pamćenja reda, deluje dosta jednostavno da se u postojeći red ubaci jedan znak – dovoljno je sledeće

tl[ey]->text[ea] = znak;

Ipak, nekoliko stvari nas sprečava da to uradimo ovako jednostavno. Prva je pitanje odvojenog prostora za red. Prošli put sam pokazao osnovnu ideju – za red se odvaja onoliko koliko stvarno teksta u njemu ima, plus neki mali višak. Kada se i višak popuni, ponovo se odvaja prostor za red, uz nekoliko praznih mesta više. Namera je da se redukuje broj zahteva za odvajanje memorije (funkcijom **malloc**) jer se time donekle ublažava problem fragmentacije memorije i dobija na brzini.

Moguće su, dakle, dve situacije. Prva je jednostavnija – kursor se nalazi u redu unutar područja koje je za red već ranije odvojeno i druga, kada je kursor van tog područja, dakle pomeren tako da „ispod" sebe nema memorijske zone koja drži red. Ovo je koncepcijsko pitanje – da li treba dozvoliti pri kretanju kursora u desno da ode i van reda, ili kursor treba zaustaviti (ili spustiti u sledeći red) kada pređe poslednji postojeći znak u redu? Drugo rešenje je, svakako, obavezno kod tekst procesora, jer je kod njega podela na redove samo nužnost – pasus je taj po kome se, u stvari, krećemo, pa je logično da odmah nakon poslednjeg znaka u tekućem redu prelazimo na sledeći red. I kod editora bi se za ovakvo rešenje mogao naći poneki dobar razlog, ali je ipak preovladalo uverenje da je bolje prvo rešenje – kursorom se može ići u desno, bez obzira na postojeći sadržaj reda, koliko god to korisnik želi. Kada odluči da unese neki znak, tek tada treba videti stvarne dimenzije reda i odvojiti potreban prostor, popunjavajući ranije nepostojeći deo reda znakom blanko.

Uz sve to, treba uzeti u obzir i postojanje **INSERT/OVERTYPE** režima rada. Režim umetanja znaka u postojeći tekst je sve češće osnovni režim rada editora, jer je daleko praktičniji. Osnovni razlog leži u činjenici da se u njemu ne može slučajno prebrisati tekst (šta sam ono beše prebrisao?), dok na njegovu jedinu manu, potrebu da se višak teksta ručno obriše, lako pristajemo, jer je nezgodno menjati osnovni režim svaki put kada treba prekucati nekoliko znakova. Sklon sam verovanju da je **OVERTYPE** režim potpuno nepotreban i da se u programima nalazi samo zbog potrebe da „editor ima sve poznate funkcije" a ne i zbog praktične vrednosti.

Pre nego što realizujemo funkciju za unos znaka u red, treba se prisetiti da kucanje nije jedini način da se znak u red unese. I sam program će interno morati da ima takvu mogućnost na raspolaganju. Najočigledniji primer je funkcija „nađi i zameni", koja treba pronađenu sekvencu da prvo ukloni iz teksta, a potom ubaci zadatu zamenu.

Listing 1

```
void movetoscreen(int x, int y, unsigned char *b, int len, int width)
{
    unsigned int _far *vss = (int _far *)0xb8000000L;
    int i;

    vss += x + y*80;
    i = 0;
    while (i<len && i<width)
        *vss++ = 0x0700 | b[i++];
    while (i<width)
    {
        *vss++ = 0x0720;
        ++i;
    }
}
```

Prikaz teksta na ekranu

Listing 2

```
main(int argc, char *argv[])
{

    initialize(8000);

    tf.refresh = REFRESH_ALL;
    for(;;)
    {
        display();
        i = getch();
        if (i==0)
            i = getch() + 256;
        switch (i) {
        case    UP: move_up(1); break;
        case  DOWN: move_down(1); break;
        case  LEFT: move_left(1); break;
        case RIGHT: move_right(1); break;
        case HOME : move_beginofline(); break;
        case END  : move_endofline(); break;
        case PGUP : move_up( tf.frameh - 1 ); break;
        case PGDN : move_down( tf.frameh - 1 ); break;
        }
        if ( i == ESC )
            break;
    }
}
```

Glavna petlja programa

Listing 3

```
#define LINESPACE(x)        (2*sizeof(int)+x)
#define OVERALLOC           4

int loadtext(char *name)
{
    FILE *f;
    char *line;
    TLP l;
    int i;

    if ( ( f = fopen(name,"r") ) == NULL)
        return 1;

    if ( ( line = malloc( MAX_LINE_LEN ) ) == NULL )
        return 2;

    while ( fgets(line,MAX_LINE_LEN,f) != NULL )
    {
        i = strlen(line);
        if (i && line[i-1] == '\n')
            line[--i] = 0;

        if ( ( l = malloc( LINESPACE( i ) ) ) == NULL )
        {
            free( line );
            fclose(f);
            return 2;
        }
        strcpy(l->text,line);
        l->len = l->alloc = i;
        tf.tl[tf.lastline++] = l;
    }
    free(line);
    fclose(f);
    return 0;
}
```

Tek da se nađe u memoriji računara

Listing 4

```
void move_down(int lines)
{
    if ( ( tf.ey += lines ) > tf.lastline )
        tf.ey = tf.lastline;
    tf.refresh = REFRESH_CURSOR;
}

void move_up(int lines)
{
    if ( ( tf.ey -= lines ) < 0 )
        tf.ey = 0;
    tf.refresh = REFRESH_CURSOR;
}

void move_left(int chrs)
{
    if ( (tf.ex -= chrs) < 0 )
        tf.ex = 0;
    tf.refresh = REFRESH_CURSOR;
}

void move_right(int chrs)
{
    if ( (tf.ex += chrs) >= MAX_LINE_LEN)
        tf.ex = MAX_LINE_LEN - 1;
    tf.refresh = REFRESH_CURSOR;
}

void move_endofline(void)
{
    if (tf.ey == tf.lastline)
        tf.ex = 0;
    else
        tf.ex = tf.tl[ tf.ey ]->len;
    tf.refresh = REFRESH_CURSOR;
}

void move_beginofline(void)
{
    tf.ex = 0;
    tf.refresh = REFRESH_CURSOR;
}
```

Kursor na sve strane

Zato ćemo izabrati opšte rešenje prema sledećem zahtevu – potrebna nam je funkcija koja će u redu **Y**, na poziciji **X** ubaciti zadati broj praznih mesta. Naravno, i njen ekvivalent, koji će iz reda izbaciti zadati broj znakova. Realizacija funkcije **insertspace** je data na slici 5. Na samom početku nalazimo na karakterističan slučaj – kursor se nalazi na poslednjem redu teksta, odnosno na prvom novom praznom redu za koji još nije odvojen prostor. Tada dodajemo potpuno novi red, ispitujući pri tome, da li smo možda udarili u limit broja redova koji smo predvideli. Ono što sledi iza ovog ispitivanja bi trebalo izdvojiti kao posebnu funkciju sa nazivom **insertemptyline**, jer će taj deo koda biti potreban i na drugim mestima. Ovde sam ga ipak ostavio unutar funkcije kako bih uštedeo nešto prostora, jer je ovaj nastavak i ovako opterećen listinzima. Dužina reda koju treba odvojiti je zadata pozicijom **X** od koje ubacujemo prostor (sve levo od **X** će biti popunjeno razmacima) dodajući na to traženi broj mesta.

Pri odvajanju prostora primetite višak definisan konstantom OVERALLOC – o razlogu je već bilo reči. LINESPACE je makro (dat na slici 3) kojim preračunavamo stvarno potreban prostor za TEXTLINE strukturu, jer ona, pored samih znakova iz reda teksta, ima i dve **int** vrednosti. Ovde se zapravo koristimo malim trikom koji je sasvim legalan u C jeziku. TEXTLINE struktura je sastavljena tako da podrazumeva da će se u njoj naći red maksimalna dužine (konstanta MAX_LINE_LEN). Uobičajeno C-ovsko **malloc(sizeof(struct TEXTLINE))** se zato ovde ne može koristiti – iako je element strukture **text** definisan kao **char** niz fiksirane dimenzije, mi mu prosto dinamičkim odvajanjem memorije skraćujemo dimenziju na stvarno potrebnu vrednost.

Ako otvaramo prostor u već postojećem redu, tada vršimo analizu mesta na kome počinje ubacivanje (da li je desno od poslednjeg postojećeg znaka), i poređenje potrebnog prostora sa onim koji je trenutno odvojen za red. Ako ustanovimo da je odvojen prostor nedovoljan, odvajamo potpuno nov red, u njega prebacuje-

Listing 5

```
int insertspace(int line, int col, int cnt)
{
    int i,j;
    int nlen;
    TLP ltmp;

    if (line>=tf.lastline)
    {
        if ( tf.lastline >= tf.maxlines )
            return 1;
        if ( (ltmp=malloc(LINESPACE(col+cnt)+OVERALLOC))==NULL)
            return 1;
        ltmp->len = col+cnt;
        ltmp->alloc = col+cnt+OVERALLOC;
        for(i=0; i < ltmp->len; ++i)
            ltmp->text[i] = ' ';
        tf.tl[ tf.lastline++ ] = ltmp;
    } else
    {
        nlen= tf.tl[ line ]->len > col ?
              tf.tl[ line ]->len + cnt : col + cnt;

        if ( tf.tl[ line ]->alloc < nlen )
        {
            if ( (ltmp=malloc(LINESPACE(nlen)+OVERALLOC))==NULL)
                return(1);

            ltmp->alloc = nlen + OVERALLOC;
            ltmp->len= tf.tl[ line ]->len;
            for(i=0; i < ltmp->len; ++i)
                ltmp->text[i] = tf.tl[ line ]->text[i];

            free( tf.tl[ line ] );
            tf.tl[ line ] = ltmp;
        } else
            ltmp = tf.tl[ line ];

        if (ltmp->len <= col)
            for(i=ltmp->len; i<col; ++i)
                ltmp->text[i] = ' ';
        else
            for(i=ltmp->len-1; i>= col; --i)
                ltmp->text[i+cnt] = ltmp->text[i];

        i=col;
        for(j=0; j<cnt; ++j)
            ltmp->text[i++] = ' ';

        ltmp->len=nlen;
    }
    return 0;
}
```

Odvajenje prostora za novi znak u redu

Listing 6

```
default  { if ( i > 0 && i<256 )
               if (!insertspace(tf.cy,tf.cx,1))
               {
                   tf.tl[ tf.cy ]->text[ tf.cx ] = i;
                   move_right(1);
                   tf.refresh = REFRESH_CURRENT;
               }
           break;
```

Funkcije za unos teksta

Listing 7

```
int removeline(int pos)
{
    int i;

    free( tf.tl[ pos ] );
    for(i=pos+1; i < tf.lastline; ++i)
        tf.tl[ i-1 ] = tf.tl[ i ];
    --tf.lastline;
}
```

Izbacivanje nepotrebnog reda

mo sadržaj starog i zatim oslobađamo memoriju koju je zauzimao.

Kada obezbedimo potreban prostor, ostaje nam samo da, ukoliko je ubacivanje unutar postojećeg teksta, sam tekst razmaknemo. Kao dodatak, može se staviti i inicijalno popunjavanje blanko simbolima. Kako se sada funkcija može primeniti, dato je na slici 6, koja predstavlja dodatak na osnovnu petlju programa. Kada pritisnuti taster spada u oblast ASCII kodova, na tekućoj poziciji kursora ubacujemo jedno prazno mesto, u njega upisujemo znak i kursor pomeramo jedno mesto desno – dakle osnovni način unosa teksta u **INSERT** režimu. U **OVERTYPE** režimu, najveći deo posla ostaje isti jer i dalje treba proveriti da li je kursor van poslednjeg reda teksta i van poslednjeg znaka u redu. Jedina razlika je što ne treba razmicati tekst, pa bi se ista funkcija, uz minimalnu modifikaciju, mogla prilagoditi i za ovaj režim prepoznajući ga, recimo, po tome što tražimo ubacivanje 0 znakova.

...I BRISANJE ZNAKA

Sve što se pojavljuje u funkciji **insertspace** koristi se i u funkciji **deletespace**, samo su ispitivanja nešto drugačije postavljene. Ako smo, recimo, na novom redu koji još ne postoji, **deletespace** ne treba da uradi ništa. Ako smo unutar postojećeg reda teksta, treba tekst desno od mesta povući u levo za zadati broj znakova. Ako nakon toga višak odvojenog prostora za red bude veći od potrebnog za više od **OVERALLOC** vrednosti, otvaramo novi red, u njega prepisujemo sadržaj tekućeg, a prostor koji je zauzimala vraćamo sistemu – dakle, sve što smo već na neki način sreli u funkciji **insertspace**.

Bitna razlika nastaje u trenutku kada brišemo znak, a kursor se nalazi odmah iza (ili bilo gde dalje) poslednjeg znaka u redu – tada dolazi do spajanja redova, ali ni ova situacija ne krije nepoznanice. Slučaj se svodi na to kao da želimo da u tekući red ubacimo **N** znakova, pri čemu je **N** broj znakova u sledećem redu. Nakon kontrole i odvajanja potrebnog prostora, što je sve viđeno u funkciji **insertspace**, treba tekst iz sledećeg reda prekopirati u otvoren prostor i zatim prosto izbaciti taj red. Način izbacivanja read je dat na slici 7.

Funkcija komplementarna ovoj – podela reda na dva dela je takođe potrebna. Ona se koristi kada korisnik u **INSERT** režimu pritisne taster **enter** u sred reda. Iza tekuće linije se sada ubacuje jedna prazna, u koju se premesta desni deo presečenog reda, a tekuća linija se samo analizira na pojavu viška prostora. Verujem da nije potrebno davati listing ovih funkcija, jer je način njihove realizacije posle iskustva za **insertspace** prilično očigledan.

SNIMANJE I UČITAVANJE

Osnovni deo funkcije za učitavanje teksta sam već dao na slici 3. Njoj nedostaje samo nekoliko detalja pa da postane kompletna. U rešenju sa slike podrazumeva se da se tekst učitava odmah nakon inicijalizacije sistema, jer se otvaraju novi redovi i pretpostavlja da je vrednost **tf.lastline** jednaka nuli. Ako editor treba da ima naredbu za učitavanje, koja se može pozvati u toku rada, pitanje je odluke da li želite da funkcija **loadtext** uvek upisuje tekst tako što će prethodno obrisati postojeći ili želite da obezbedite funkciju za preklapanje – svako učitavanje ubacuje novi tekst sa diska tako što umeće redove od mesta na kome se nalazio kursor. Ovo drugo rešenje je opštije jer se i prvi slučaj svodi na njega, ako se prethodno postojeći tekst obriše. Zato funkcija **loadtext** umesto tela petlje sa slike 3 može koristiti funkciju **insertemptyline**. Ako se, dakle, samo pozove **loadtext**, dobijamo funkciju **merge** – umetanje teksta sa diska u postojeći tekst. Ako prethodno obrišemo sav postojeći tekst i postavimo kursor na poziciju 0,0, pa potom pozovemo **loadtext** – imamo čisto učitavanje novog teksta.

Snimanje teksta je više nego trivijalno – treba krenuti od prve ka poslednjoj liniji i na disk snimati sadržaj redova, dodajući im prethodno znak **novi red** na kraj.

POGLED IZNUTRA

Verujem da ste nakon ovog nastavka stekli sliku o tome kako izgleda jedan editor iznutra. Možda bi se cela suština mogla opisati jednom rečju – stalno premeštanje blokova memorije sa jednog mesta na drugo. Ili, kako bi neko koga sve ovo ne zanima, mogao primetiti – presipanje iz šupljeg u prazno! Ipak, editor ne čini samo ova osnova, već i niz onih funkcija koje mu daju dodatnu upotrebnu vrednost vezanu za specifične poslove. Ono što na prvi pogled nije očigledno, ali će vam sigurno pasti na pamet ukoliko se upustite u izradu editora, jeste da ova osnova predstavlja idealnu bazu za nadgradnju programa koji nisu samo editori. Sve što nam treba za brzu manipulaciju tekstom je tu – funkcije da ga dovedemo u memoriju računara, funkcije za promenu sadržaja, funkcije za snimanje na disk. To što ima nekoliko funkcija koje se bave prikazom na ekranu ne predstavlja poseban višak. Treba zameniti kodni raspored u nekom tekstu, treba ga preformatirati po estetskim kriterijumima izvornog koda nekog jezika, treba proveriti ispravnost napisanih reči... mogućih upotreba je zaista puno.

U sledećem broju, u poslednjem tekstu iz ove serije, pozabavićemo se funkcijama za manipulaciju blokovima teksta, jer se radi o najkomplikovanijem delu, koje ima puno mogućih rešenja. Nekoliko reči ćemo posvetiti i ostalim interesantnijim funkcijama iz neobaveznog skupa.

Ovog meseca bistrimo interapte i izlazne procedure. Videćemo da posebnim komandama te sa nešto malo znanja o sistemskim rutinama možemo kreirati kakvu bilo obradu prekida, a da putem izlaznih procedura možemo kontrolisati zatvaranje naših aplikacija bez obzira na uzrok koji je doveo do kraja izvršavanja. Da sve to što sledi nije prazna priča dokazuje i priloženi program u *Turbo Pascal*-u 6.0: demonstrira najefektniju primenu interapta i kontrole toka – multitasking!

Upravljanje interaptima

Još je u ranim verzijama bilo omogućeno korisniku da upravlja interaptima (prekidima u izvršavanju programa) i da u najobičnijem *TurboPascal* piše procedure za njihovu obradu. U novim *TurboPascal*-u 6.0 i *TurboPascal for Windows* je ovaj posao dodatno unapređen ubacivanjem integrisanog asemblera – o kojem smo pisali prošlog meseca – čime se otvara mogućnost za lako i jednostavno pisanje brzih interapt procedura u čistom asembleru a u okviru listinga sa glavnim programom. Potreba za interaptima je dosta česta u praksi, a procesi koje treba obavljati kontinuirano ali ukorak sa glavnim tokom mogu biti veoma različite prirode, od toga da vam računar svira *Fraire Jacques* dok kucate neki tekst, preko sprajt rutina do složene kontrole raspodele memorije, komunikacije sa periferijama, servisnih prekida, multitaskinga i slično.

Procedura za obradu prekida se deklariše rezervisanom reči **Interrupt**. Ona se umeće neposredno iza identifikatora procedure i navodi kompajler da pri njenom prevođenju generiše sledeći osnovni ulazni i izlazni kod:

```
PUSH AX     itd       POP SI
PUSH BX               PUSH ES
MOV SP BP             POP DX
PUSH CX               PUSH BP
POP BP                POP CX
PUSH DX               MOV BP SP
POP ES                POP BX
PUSH SI               SUB SP LocalSize
POP DS                POP AX
PUSH DI               MOV AX DSeg IRET
POP DI                MOV DS AX
PUSH DS                 itd
```

Kao što vidite, na početku se svi registri stavljaju na stek, jer se prekid može dogoditi praktično na bilo kom mestu u glavnom toku. Registar BP preuzima vrednost stek pointera, SP se spušta za veličinu lokalnih varijabli. Inicira se DS na **DataSegment** jer se tamo nalaze varijable koje koristi interapt procedura – isto kao i svaka druga. Osnovni ulazni i izlazni kod izgleda nešto drugačija kada se interapt procedura prevodi opcijom **$G+** za prošireni set instrukcija 286-ice: na početku i na kraju se umeću ENTER i LEAVE.

U zaglavlju se mogu navesti parametri (svi ili samo neki, počev od nekog pa nadalje redom bez izostavljanja) tipa **word** koji će preuzeti vrednosti registara zatečene u trenutku prekida, tako da ih možemo ispitivati i menjati:

IntProc(Flags,CS,IP,AX,BX,CX,DX,SI,DI,DS,ES,BP word),

Primetite da nema registara SP i SS zato što za njih već postoje funkcije **SPtr** i **SSeg**. Pored toga, u pomenute dve najnovije verzije moguće ih je očitavati i menjati vrednosti registara bez neke posebne petljancije, kako u listingu glavnog programa, tako i u interapt procedurama. Mogućnost prethodne deklaracije je u njima zadržana jedino zbog kompatibilnosti sa ranijim verzijama.

Veoma važno: interapt procedura ne sme imati lokalne varijable deklarisane sa **var**!

Raspoložive komande

E sad, samo time što smo kreirali interapt proceduru i deklarisali je kao takvu, njeno izvršavanje neće automatski početi. Nužno je iduči ovim redom: odabrati određeni interapt vektor, privremeno mu sačuvati vrednost u nekoj pokazivačkoj varijabli, usmeriti ga željim na našu proceduru, izvršavati je neko vreme i na posletku vratiti staru vrednost vektora. Zavisno od toga koji smo vektor odabrali, da li se pobuđuje hardverski ili softverski, prekidi se mogu obrađivati kontinuirano u jednakim vremenskim intervalima (kod PC-ja je to oko 1/18 sec) odnosno jednom po pozivu, nalik DOS ovom **INT $21**. Imamo na raspolaganju sledeći set komandi:

- **SetIntVec(IntNo,IntProcPtr)** postavlja vektor prekida broj **IntNo** (tip **byte**) na korisnički definisanu proceduru na koju pokazuje pointer **IntProcPtr**
- **GetIntVec(IntNo,Pointer)** puni **Pointer** adresom procedure za obradu prekida na koju je usmeren **IntNo**, kao i gore tipa **byte**
- **Intr(IntNo Regs)** zaustavlja program i izvršava jedanput **IntNo** softverski prekid. Slog **Regs** je tipa **Registers**, koji je deklarisan unutar jedinice DOS, a sadrži deset polja (**AX BX CX DX, BP SI DI DS ES** i **Flags**) tipa **word**. Polja se pune vrednostima odgovarajućih registara zatečenih u trenutku prekida, pa im procedura za obradu interapta lako može pristupati. Eventualno vraćanje rezultata obavlja se takođe putem ovog sloga – pravi registri, međutim, neće biti istovetni sa novim vrednostima polja, jer se oni obnavljaju nizom POP instrukcija izlaznog koda
- **MsDos(Regs)** radi isto što i **Intr**, samo se ne navodi broj interapta jer se podrazumeva DOS ov servisni prekid **$21**. A i funkcija sloga **Regs** je ovde unekoliko drugačija: **Intr** puni polja sloga vrednostima registara dok suprotno, MSDOS upisuje redom u odgovarajuće registre sadržaj polja (jasno je zašto ne postoje polja za SS i SP). Na ovaj način se servisnom prekidu kroz registre prosleđuju parametri.
  Novija alternativa poslednjim dvema komandama je kratka **asm** sekvenca u njoj željenim registrima dodelimo vrednosti i izvršimo instrukciju INT
- **SwapVectors** procedura zamenjuje vrednosti seta pointera **SaveIntXX** iz jedinica **System** sa trenutnim pokazivačima vektora. **SwapVectors** se umeće neposredno ispred i iza poziva (putem **Exec** naredbe) nekog spoljnjeg programa, onda kada nismo sigurni da nam on neće zabrljati naše vektore prekida. Postoji ukupno 18 takvih pointera, ovo **XX** iz **SaveIntXX** može biti 00, 02, 1B, 23, 24, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F ili 75, sve heksadecimalno. Jedinica **System** pre nego što postavi sopstvene vektore prekida puni ovaj set pointera starim vrednostima. Na izlasku iz aplikacije, standardna izlazna rutina učini obrnuto.

Treba reći i to da pri pravljanju vaših prekidnih procedura uopšte niste obavezni oslanjati se isključivo na deklarisanje sa **interrupt**, naprotiv, napredniji programer će pre u asembleru napisati kompletnu proceduru sa svojim ulaznim i izlaznim kodom, izbegavajući tako stalno stavljanje na stek registara koji se u njoj ne menjaju. Naredbe **SetIntVec** i **GetIntVec** nam to sa svoje strane omogućuju, jer ne gledaju da li je procedura iz parametra deklarisana kao prekidna ili nije.

Izuzetno je važno ne dozvoliti interapt proceduri da dođe u koliziju sa glavnim tokom. S tim u vezi se ne preporučuje da ona koristi **heap** komande za dinamičku alokaciju memorije, DOS pozive, druge interapte, spoljne programe itd. Trebalo bi se čuvati i oslanjanja na potprograme kojima se služi i glavni tok, ako ovo nije moguće izbeći, treba barem parovima mašinskih instrukcija CLI/STI uokviriti (najzgodnije putem **Inline**) takve delove programa.

Izlazne procedure

Očigledna je potreba za mehanizmom sigurnog izlaska iz programa, a nezavisno od načina njegovog završetka, npr. ako smo instalisali sopstvene vektore prekida, treba pri povratku u DOS obavezno vratiti predašnje, ako naša aplikacija upravlja sa datotekama, uvek mora na kraju rada zapitati nešto u stilu „Save changes Y/N". Takođe, pri izlasku valja vratiti prvobitne atribute slova, brzinu *autorepeat*-a tastera, oblik kursora... znači sve ono što se u aplikaciji menjalo.

U **Turbo Pascal**-u takav mehanizam postoji i realizuje se nekom vrstom dinamičke jednostruko povezane liste pointera na korisnički definisane izlazne procedure. Lista tih procedura može neprestano da se ažurira u toku rada aplikacije – samim tim što se i protokol njenog zatvaranja menja u zavisnosti od mesta na kojem je došlo do ispadanja. Uobičava se u praksi da svaka biblioteka/modul koja otvara novi proces na sistemskom nivou ima sopstvenu izlaznu proceduru koju uključuje u listu za vreme svoje aktivnosti.

Glavnu ulogu u svemu ovome igra pokazivačka varijabla **ExitProc** jedinice **System**: prima početnu adresu procedure koju je obavezno izvršiti pre napuštanja programa, bez obzira da li se do kraja došlo strukturirano (sa end.), preko **Halt** naredbe, sa Ctrl Break ili je pak došlo do *run time* greške. Predefinisano **ExitProc** na startu ukazuje na standardni izlazni kod koji ima ulogu da pozatvara **Input** i **Output** datoteke i restaurira interne vektore prekida samog **Turbo Pascal**-a. Programer može da izmeni **ExitProc** tako da pokazuje na njegovu izlaznu proceduru, s tim da prethodno pohrani prvobitnu vrednost u neku pomoćnu pokazivačku promenljivu. Prvo što bi onda morala da uradi ta procedura jeste da vrati raniju vrednost jer bi u suprotnom mogli nastati veliki problemi: pošto izlazna rutina na kraju svake jedinice, koja i vrši iščitavanje **ExitProc** i pozivanje te adrese, pre nego što prepusti kontrolu korisnikovoj izlaznoj proceduri setuje **ExitProc** na **nil** (što je ispravno, s obzirom da bi se u slučaju *run time* greške unutar nje ona opet pozvala, te bi nastupila mrtva petlja), dogodilo bi se, dakle, da lista od više izlaznih procedura bude prekinuta! Naime, čim izlazna rutina konstatuje da je **ExitProc** jednako **nil**, prelazi na izvršavanje sopstvenog izlaznog koda.

Poziv izlaznoj proceduri se vrši instrukcijom **CALL Seg:Ofs**, pa zato ona mora biti prevedena sa **$F+** da bi funkcionisao FAR model pozivanja.

Listing 1 MAIN.PAS

```
{$F+}               { Glavni program MAIN   povezuje jedinice }
                    { MTasking, SortTask i ocitava tastaturu  }
program  MAIN;      { Miroslav Olanjin (c)  Mart/April 1992   }

uses     Crt, MTasking, SortTask;

begin    { program }
  ClrScr;
  Randomize;
  repeat
    if KeyPressed then begin
      case (ReadKey) of
        #00 : case (ReadKey) of
                #59 : InitNewTask(BubbleSort);       { F1 }
                #60 : InitNewTask(SelectionSort);    { F2 }
                #61 : InitNewTask(ShellSort);        { F3 }
              end;
        #27   begin TextColor(7); ClrScr; Halt; end; { ESC }
      end;
    end;
  until false;
end      { program }
```

Glavni program MAIN – povezuje jedinice *MTasking*, *SortTask* i očitava tastaturu

Listing 2 MTASKING.PAS

```
{$F+}                     { Jedinica za upravljanje paralelnim  }
                          { izvrsavanjem do 4 razlicita procesa }
unit   MTasking;          { Miroslav Olanjin (c) Mart/April 1992 }

INTERFACE

uses   Dos;

type   Proc = procedure;
       TaskType = record
                    Execute     Boolean;
                    Active      Boolean;
                    StackPtr    word;
                    TProc       Proc;
                  end;

const  StackTop    array [1..3] of word = (32767,49151,65535);

var    Tasks                          array [0..3] of TaskType;
       OldVect, OldExit               pointer;
       FreeTask, RunTask              word;
       StartTask, NewStack            word;
       Word1, Word2, Word3, Word4     word;

procedure InitNewTask (NewTask : Proc);


IMPLEMENTATION


procedure InitNewTask (NewTask : Proc);
begin
  for FreeTask :=1 to 3 do begin
    with Tasks[FreeTask] do begin
      if not(Execute and Active) then begin
        Execute :=true;
        TProc :=NewTask;
        Exit;
      end;
    end;
  end;
end;

procedure EndOfTask (TaskNumber : word);
begin
  Tasks[TaskNumber].Execute :=false;
  Tasks[TaskNumber].Active :=false;
  asm INT $08 end;
end;


procedure TaskManager; interrupt;
begin
  Tasks[RunTask].StackPtr :=SPtr + 4;
  for StartTask :=1 to 3 do begin
    with Tasks[StartTask] do begin
      if Execute and not(Active) then begin
        Active :=true;
        RunTask :=StartTask;
        NewStack :=StackTop[StartTask];
        Word1 :=Seg(EndOfTask); Word2 :=Ofs(EndOfTask);
        Word3 :=Seg(Tproc); Word4 :=Ofs(Tproc);
        asm
          MOV     SP,NewStack      { Iniciraj novi vrh steka }
          PUSH    StartTask        { Parametar za EndOfTask  }
          SUB     SP,04            { Mesto za LocalSize var. }
          PUSH    Word1            { Pohrani na stek segment }
          PUSH    Word2            { i ofset od EndOfTask    }
          PUSHF                    { Stavi flegove zbog IRET }
          PUSH    Word3            { Pohrani segment i ofset }
          PUSH    Word4            { adrese procesa i javi   }
          MOV     AL,$20           { inter. kontroleru 8259  }
          OUT     $20,AL           { za kraj obrade prekida  }
          IRET                     { = POP IP, POP CS, POPF  }
        end;
      end;
    end;
  end;
  repeat
    if RunTask = 3 then RunTask :=0 else inc(RunTask);
  until Tasks[RunTask].Active;
  NewStack :=Tasks[RunTask].StackPtr;
  asm
    MOV AL,$20                     { Zatvori obradu prekida  }
    OUT $20,AL                     { broj 08H, izlazni kod   }
    MOV SP,NewStack                { obnavlja SP preko BP    }
  end;
end;


procedure CloseIntExit;
begin
  SetIntVec($08,OldVect);
  ExitProc :=OldExit;
end;


begin   { unit }
  for FreeTask :=1 to 3 do begin
    Tasks[FreeTask].Execute :=false;
    Tasks[FreeTask].Active :=false;
  end;
  RunTask :=0;
  Tasks[0].Active :=true;
  OldExit :=ExitProc;
  ExitProc :=@CloseIntExit;
  GetIntVec($08,OldVect);
  SetIntVec($08,@TaskManager);
end.    { unit }
```

Jedinica za upravljanje paralelnim izvršavanjem do 4 različita procese

Ako dalja akcija u vašoj izlaznoj proceduri treba da zavisi od vrste uzroka okončanja programa, obratite pažnju na sistemske varijable **ExitCode** (integer) i **ErrorAddr** (**pointer**) obe iz jedinice **System**. U slučaju da je program strukturano došao do kraja, **ExitCode** će biti nula a **ErrorAddr nil**. Ako je prethodila naredba **Halt(n)**, **ExitCode** će imati vrednost parametra **n** ili nulu ako **n** nije bio naveden. Ispad u slučaju *run time* greške puni **ExitCode** kodom te greške, a **ErrorAddr** adresom problematičnog iskaza. Evo kodova grešaka koje najčešće dovode do prekida izvršavanja:

200 – Division by zero
201 – Range check error
202 – Stack overflow error
203 – Heap overflow error
204 – Invalid pointer operation
205 – Floating point overflow
207 – Invalid floating point operation

Turbo Pascal for Windows pored ovih predviđa još i sledeće fatalne greške:

210 – Object not initialized
211 – Call to abstract method
212 – Stream registration error
213 – Collection index out of range
214 – Collection overflow error

Pod *Windows* okruženjem *run time* greška uzrokuje zatvaranje svih prozora što ih je inicirala aplikacija, oslobađanje memorije koju je zauzimala i ispisivanje poruke od strane operativnog sistema. Isto kao u DOS-u, ako ne dozvolite da **ExitCode** sadrži broj različit od nule, izbeći ćete intervenciju operativnog sistema i privešćete aplikaciju kraju kao da nije bilo posredi *run time* greške!

Multitasking

Kako u literaturi već postoje primeri jednostavnijih interapt procedura, opredelili smo se da damo jedan relativno složeniji program za koji lako možete naći i konkretnu primenu u nekoj vašoj aplikaciji. Na listingu 2 se nalazi mala jedinica **MTasking** za upravljenje paralelnim izvršavanjem do 4 različita procese, što će reći da omogućava multitasking. Multitasking se ostvaruje tako što CPU ciklično skače sa jednog na drugi proces dovoljnom učestalošću da nam se čini kako se oni odvijaju istovremeno.

Iz listinga se vidi da u INTERFACE delu jedinice **MTasking** postoji samo jedna komanda za komunikaciju sa jedinicom: procedura **InitNewTask(ProcVar)** pretražuje tabelu procesa **Tasks**, pa ako ih je u tom trenutku pokrenuto manje od četiri, inicira otvaranje novog **ProcVar** procesa. Glavni blok, koji se odredi odmah pri startovanju aplikacije, usmerava interapt vektor broj 08H (Timer interrupt) na pol-

Listing 3 SORTTASK.PAS

```
{$F+}               { Jedinica  biblioteka od tri procesa    }
                    { predvidjenih za paralelno izvrsavanje  }
unit   SortTask;    { Miroslav Glanjin (c)  Mart/April 1992  }

INTERFACE

uses   Crt, Dos;

const  MaxDuz = 20;

type   TSortServis = object
            Niz : array [1..MaxDuz] of word;
            procedure Init(poz:integer);
            function  Swap(el1,el2,poz:integer):Boolean;
            procedure ShowBar(el,poz:integer);
            procedure CleanPad(poz:integer);
       end;

procedure BubbleSort;
procedure SelectionSort;
procedure ShellSort;


IMPLEMENTATION

  procedure TSortServis.Init(poz:integer);
  var  n1 : integer;
  begin
    for n1 :=1 to MaxDuz do begin
      Niz[n1] :=Random(18)+2;
      ShowBar(n1,poz);
    end;
  end;

  function TSortServis.Swap(el1,el2,poz:integer):Boolean;
  var  Temp : word;
  begin
    If Niz[el1] > Niz[el2] then begin
      Delay(50);
      asm CLI end;
      Temp :=Niz[el1];
      Niz[el1] :=Niz[el2];
      Niz[el2] :=Temp;
      ShowBar(el1,poz);
      ShowBar(el2,poz);
      Swap :=true;
      asm STI end;
    end else Swap :=false;
  end;

  procedure TSortServis.ShowBar(el,poz:integer);
  var  n1, n2 : integer;
  begin
    asm CLI end;
    GotoXY(poz,el+2); TextColor(15);
    for n1 :=1 to Niz[el] do Write(Chr(220));
    TextColor(0);
    for n2 :=n1 to MaxDuz do Write(Chr(220));
    GotoXY(poz,el+2); TextColor(15);
    asm STI end;
  end;

    procedure TSortServis.CleanPad(poz:integer);
    var n1, n2 : integer;
    begin
      asm CLI end;
      for n1 :=1 to 24 do begin
        GotoXY(poz,n1);
        for n2 :=poz to poz+20 do Write(' ');
      end;
      GotoXY(1,1);
      asm STI end;
    end;


  procedure BubbleSort;
  var  BubS     : TSortServis;
       n1, n2   : integer;
       Bubble   : Boolean;
  begin
    BubS.Init(8);
    GotoXY(8,1); Write('Proc1: BubbleSort');
    n1:=MaxDuz-1;
    repeat
      for n2 :=1 to n1 do begin
        Bubble :=BubS.Swap(n2,n2+1,8);
      end;
      Dec(n1);
    until (n1 = 0);
    BubS.CleanPad(8);
  end;


  procedure SelectionSort;
  var  SelS       : TSortServis;
       n1, n2     : Integer;
       Selection  : Boolean;
  begin
    SelS.Init(31);
    GotoXY(31,1); Write('Proc2: SelectionSort');
    for n1 :=1 to MaxDuz do begin
      for n2 :=n1+1 to MaxDuz do begin
        Selection :=SelS.Swap(n1,n2,31);
      end;
    end;
    SelS.CleanPad(31);
  end;


  procedure ShellSort;
  var  ShellS              : TSortServis;
       n1, n2, s1, s2      : Integer;
       Shell               : Boolean;
  begin
    ShellS.Init(56);
    GotoXY(56,1); Write('Proc3: ShellSort');
    s1 :=MaxDuz;
    while (s1 > 1) do begin
      s1 :=s1 div 2;
      n2 :=MaxDuz-s1;
      repeat
        Shell:=false;
        for n1:=1 to n2 do begin
          s2 :=n1 + s1;
          If ShellS.Swap(n1,s2,56) then Shell :=true;
        end;
      until not(Shell);
    end;
    ShellS.CleanPad(56);
  end;
  end    { unit }
```

Jedinica – biblioteka od tri procesa predvidjenih za paralelno izvršavanje

program *TaskManager* neku vrstu skretnice za upravljanje procesima. *TaskManager* u pravilnim vremenskim intervalima prekida glavni tok i ispituje tabelu procesa. Ako nađe da je u nekom od 4 sloga polje **Execute** jednako **true** a da taj proces nije već u toku (**Active** **false**) postavlja novu početnu vrednost *stek pointera*, pohranjuje na stek punu adresu procedure procesa kojeg treba inicirati (zbog IRET instrukcije), zatim adresu procedure **EndOfTask** te njen parametar, redni broj upravo pokrenutog procesa. Time se na izlasku iz obrade interapta kontrola predaje novom procesu. On će teći sve do sledećeg prekida, kada će **TaskManager** vratiti stari SP, od njegove pozicije iscitati ondašnje vrednosti registara i vratiti kontrolu glavnom toku ili nekom drugom procesu. Kada neki od procesa dođe do kraja, IRET instrukcija na kraju njegovog koda predaje kontrolu završnoj **EndOfTask** proceduri, koja će zatvoriti proces upisivanjem **false** u polja **Active** i **Execute** i softverskim pozivanjem *TaskManager*-a.

Jedinica **MTasking** se brine i o vraćanju interapt vektora koji su važili pre nje. **ExitProc** pokazivačka sistemska varijabla usmerava se na proceduru **CloseIntExit** i ona će se odraditi bez obzira na način završetka rada programa. Čitalac može prema sopstvenim potrebama nadograđivati jedinicu **MTasking**: prvo, može unutar asemblerskog dela procedure *TaskManager* ubaciti instrukciju **CALL OldVec8** kojom bi se pri svakom prekidu odradila i ranija rutina za obradu tog interapta; drugo, možete definisati i posebne izlazne procedure za svaki od procesa, kako bi se u slučaju kritične greške izašlo samo iz dotičnog procesa a ne iz celokupne aplikacije. U tu svrhu bi trebalo dopuniti proceduru *TaskManager* tako da ona pri predaji kontrole drugom procesu izmeni i pokazivač na poslednju izlaznu proceduru u listi, a staru vrednost zapamti u dodatnom polju sloga **TaskType**.

Kao nekakvi procesi koji se paralelno izvršavaju uzeta su tri algoritma za sortiranje **BubbleSort**, **SelectionSort** i **ShellSort**. Njihove implementacije čine jedinicu **SortTask** listing 1. Ne slučajno su iz familije sličnih algoritama izabrani baš oni sporiji, da bi se moglo lepo posmatrati kako rade uporedo. Da je uzet recimo *QuickSort*, sve bi se obavilo dok kažete keks.

Obratite pažnju da svi algoritmi koriste nekoliko zajedničkih rutina; da ne bi došlo do zajedničke upotrebe i internih varijabli u njima (što bi bilo veoma rizično), pozvani su u pomoć objekti: svaki objekat sebi pridružuje i interne promenljive deklarisane unutar metoda. Klasa **TSortServis** sastavljena je iz tipa niza koji se sortira i metoda **Init**, **Swap**, **ShowBar** i **ClearPad**. Svaki od procesa na osnovu ove klase inicira sopstveni objekat: **BubbleSort** procedura incira **BubS**, **SelectionSort** **SelS**, a **ShellSort** **ShellS** objekat.

Konačno, glavni program *Main* sa listinga 3 povezuje obe jedinice, ispituje tastere F1, F2 i F3 i u zavisnosti da li je neki od njih pritisnut pokreće izvršavanje određenog procesa.

488 [OS OS/2, PC] Ova rubrika je hvala Bogu i IBM-u tokom aprila dobila novu, značajnu i izuzetno zanimljivu temu kojom će se kako izgleda dosta baviti tokom sledećih meseci i godina – OS/2 2.0 je stigao i već u svojim prvim danima vrodvao brojne pristalice. Naravno, nije da nije bilo i dosta problema, naročito oko instalacije. Događa se lako da posle nekih neopreznih intervencija na (po obimu povećim) konfiguracionim datotekama sistem na prosto odbije da se podigne. U tom slučaju pokušajte da tokom butovanja, kada se pojave tri kvadrata sa OS/2 znakom pritisnete Alt F1 a zatim te tastere ne puštajte sledećih dvadesetak sekundi. Na taj način biće reaktivirana originalna verzija CONFIG.SYS-a koju ponovo možete menjati.

Ako ni to ne pomaže umetnite prvu OS/2 disketu (*Installation*) resetujte računar i kada se pojavi tekstualni OS/2 logo pritisnite Esc. Rezultat je skoro kao da ste pod DOS-om podigli sistem sa DOS diskete – još kada biste imali editor. [Preuzeto sa Sezama, autor **Dragan Selaković** (*dselakovic*)]

489 [OS OS/2, OTXT WPERF, PC] Od operativog sistema nema mnogo koristi ako nemate programe za njega, a za početak će poslužiti i DOS odnosno *Windows* aplikacije. Na primer *WordPerfect for Windows* – propisno ga instalirala njegov *file manager*, *thesaurus* i druge sitnice lepo rade ali sam *WordPerfect* ni makac! Rešenje donosi README fajl samog OS/2. Treba preći u direktorijum OS2 i startovati program FIXWP koji pogađate „sredi" *WordPerfect for Windows*. Da bi čitava stvar bila još zanimljivija, ovako „popravljeni" *WordPerfect for Winbows* radi neuporedivo pouzdanije nego originalna verzija na *Windows 3.0* – nije nam ni jednom uspelo da ga „srušimo"! [Preuzeto sa Sezama, autor **Pavle Peković** (*ppekovic*)]

490 [OS MSDOS, PC] OS/2 je OS/2, ali i o DOS-u nismo sve rekli. Elem, kažu da je *Unix* odiji žalo što ima naredbe PUSHD i POPD – nalazite se u nekom direktorijumu, kažete PUSHD, odete u neki drugi, uradite šta imate i sa POPD se vratite u prethodni. Ima PD programčića koji rade slične stvari ali evo i jednog *batch* fajla (slika 1) – ne samo da je praktičan nego ćete imati prilike da upoznate i trikove primenjene u te dve linije. [Autor *Arnell P. Reyes, New Mexico*, preuzeto iz PC Magazine-a]

491 [PJ TPASCAL, PC] Jeste li isprobali primer zvani EDITORS.PAS iz *Turbo Pascal*-a 6.0? Ako ste pri tome bili pažljivi, svakako ste primetili da se brojevi 3-6 uopšte ne mogu otkucati! Evo *patch*-a koji to otklanja: učitajte EDITORS.PAS u Turbo editor i zamenite originalne takstove ispravkama u skladu sa slikom 2. Ponovo kompajlirajte program i sve će biti u redu! [Preuzeto sa BIX-a, zvaničan *Borland*-ov materijal]

492 [PJ TURBOC, PC] Kako sa ekrana skloniti kurzor? Na *Turbo C*-u ne bi trebalo da bude teško: na raspolaganju je funkcija *_setcursortype* tako da se sa *_setcursortype (_NOCURSOR)* kurzor uklanja, sa *_setcursortype (_NORMALCURSOR)* vraća „u normalu" dok *_setcursortype (_SOLIDCURSOR)* pretvara kurzor u pun pravougaonik. Ukoliko radite u nekom drugom jeziku, zanimaće vas univerzalno rešenje – stavite 1 u AH, 20h u CH i nulu u CL pa pozovite INT 10. [Preuzeto sa Sezama, autori **Dušan Popović** (*dusanp*) i **Janko Stamenović** (*janko*)]

493 [KOMS, PC] Posvećeno ljubiteljima *Nortonovih* programa: jeste li nekada poželeli da imate *Norton Guide* za *Norton Commander 3.0*? Jeste? Želja vam je ispunjena i pre nego što ste je izrekli: pronađite u *Norton Commander*-ovom direktorijumu datoteku NC.HLP, prekopirajte je u NC.NG i – eto *Norton Guide*-a! [Preuzeto sa Sezama, autor **Nikola Bošković** (*nboskovic*)]

494 [COMM] Ukoliko nemate *email* adresu nekog poznanika u Evropi a želite da sa sa njim (ili njom) dopisujete, pošaljite na adresu EARN: NETSERV@xxxxxxxx.BITNET poruku u kojoj piše *UDS GET [broj] NAME ime* ili *UDS GET [broj] ADDR adresa* ili *UDS GET [broj] PHONE telefon* ili neku kombinaciju, npr *UDS GET [broj] NAME ime ADDR adresa*.

Da vidimo šta je šta u ovim „rebusima": *xxxxxxxx* označava odredište na osnovu tabele sa slike 3. Ime je bilo koji niz (najviše 52 znaka) koji se nalazi u imenu željenog korespondenta, adresa je niz slova koji se pojavljuje u njegovoj/njenoj adresi a *telefon* deo telefona. Parametar *broj* je opcioni i omogućava da ograničite koliko će vam imena koja zadovoljavaju zadate uslove sistem poslati.

Nekoliko minuta ili sati kasnije dobićete poruku sa svim korisnicima EARN-a koji se uklapaju u zadatu specifikaciju i njihovim adresama.

Ukoliko ne znate nikoga ali želite da se dopisujete sa ljudima koji se bave sličnim stvarima, odaberite neku od nabrojanih zemalja i na opisanu adresu pošaljite poruku u kojoj, na primer, piše *UDS GET [broj] ADDR grad DESCR osobine* – dobićete spisak svih stanovnika zadatog grada koji se interesuju za zadate osobine. *UDS GET 25 ADDR Paris DESCR music*, na primer, daje spisak svih korisnika EARN-a iz Pariza koje zanima muzika. [Preuzeto sa DECnet-a, autor **Marinković** (*UBBG EMARINKA*)]

495 [COMM 494, PC] Opisali smo kako da pronađete druge, a sada da vidimo kako da drugi pronađu vas! Pošaljite na adresu EARN: "NETSERV@AEARN.BITNET" poruku u kojoj piše

UDS ADD
:USERID xxxxxxxx :NODE yyyyyyyy
:NAME ime prezime
:PHONE telefon
:ADDR ime_institucije, grad, zemlja
:DESCR profesija, aktivnosti, interesovanja,

xxxxxxxx i *yyyyyyyy* su vaše korisničko ime odnosno oznaka računara sa koga zovete (najviše po 8 znakova), ime i prezime mogu da budu dugački 52 znaka kao i broj telefona, adresa i lista profesija odnosno interesovanja. Ako ste podatke već poslali, možete ih modifikovati komandom UDS REP. [Preuzeto sa DECnet-a, autor **Marinković** (*UBBG EMARINKA*)]

496 [UTIL PKLITE, PC] Mnogi programeri koriste *shareware* programčić PKLITE ne da bi skratili svoje EXE datoteke nego da bi „sakrili" tekstualne poruke u njima i tako ih zaštitili od neovlašćenih izmena. Ovakva zaštita na žalost, nije mnogo efikasna pošto PKLITE „ume" da raspakuje ono što je zapakovao i tako vrati program u početno stanje. Vlasnici registrovanog PKLITE-a mogu da pokušaju sa opcijom

Listing 1

```
@IF %1 == X PROMPT $$$:$ $CD $P
@IF %1'==' %COMSPEC% /C %0 X > c:\utils\popd.bat
```

Povratak u prethodno aktivni direktorijum (POPD)

Listing 2

```
Stoji:                          Treba:

      LODSW                           LODSW
      OR      AL,AL                   CMP     AL,DL
      JE      @@2
      CMP     AL,DL

      JE      @@4                     JE      @@4
@@2:  CMP     DH,DH                   CMP     DX,DX
      JE      @@4                     JE      @@4

var                             var
  Key: Word;                      ShiftState: Byte absolute $40:$17;
                                  Key: Word;

begin                           Begin
  Key := Event.KeyCode;           if (ShiftState and $03<>0) and
                                     (Event.ScanCode >= $47) and
                                     (Event.ScanCode <= $51) then
                                     Event.CharCode := #0;
                                   Key := Event.KeyCode;
```

Patch za datoteku EDITORS.PAS

Listing 3

Austrija	AEARN
Belgija	BEARN
Čehoslovačka	CSEARN
Engleska	UKACRL
Finska	FINHUT
Francuska	FRMOP11
Grčka	GREARN
Holandija	HEARN
Irska	IRLEARN
Italija	ICNUCEVM
Izrael	TAUNIVM
Mađarska	HUEARN
Nemačka	DEARN
Norveška	NORUNIT
Poljska	PLEARN
Švajcarska	CEARN
Španija	EBOUB011
Švedska	SEARN
Turska	TREARN

Odredišta u *e-mail* adresi

Listing 4

```
SET EXACT OFF
        "ABC"     =   "ABC"      - TRUE
        "ABC"     =   "ABCDE"    - FALSE
        "ABCDE"   =   "ABC"      - TRUE
        "ABC"     =   "ABC  "    - FALSE
        "ABC  "   =   "ABC"      - TRUE
        "ABC"     =   "  ABC"    - FALSE
        "  ABC"   =   "ABC"      - FALSE
        "ABC"     =   ""         - TRUE
        ""        =   "ABC"      - FALSE

SET EXACT ON
        "ABC"     =   "ABC"      - TRUE
        "ABC"     =   "ABCDE"    - FALSE
        "ABCDE"   =   "ABC"      - FALSE
        "ABC"     =   "ABC  "    - TRUE
        "ABC  "   =   "ABC"      - TRUE
        "ABC"     =   "  ABC"    - FALSE
        "  ABC"   =   "ABC"      - FALSE
        "ABC"     =   ""         - FALSE
        ""        =   "ABC"      - FALSE

SET EXACT ON ili OFF
        "ABC"     ==  "ABC"      - TRUE
        "ABC"     ==  "ABCDE"    - FALSE
        "ABCDE"   ==  "ABC"      - FALSE
        "ABC"     ==  "ABC  "    - FALSE
        "ABC  "   ==  "ABC"      - FALSE
        "ABC"     ==  "  ABC"    - FALSE
        "  ABC"   ==  "ABC"      - FALSE
        "ABC"     ==  ""         - FALSE
        ""        ==  "ABC"      - FALSE
```

Operatori poređenja u *Clipper*-u

-E, ali svima je na raspolaganju nešto mnogo jednostavnije: ako pomoću nekog disk editora (npr. *Norton Utilities*) pogledate zaglavlje komprimovane datoteke, videćete da tamo piše PKLITE. Bilo kakva izmena tog teksta neće uticati na normalan rad programa ali će onemogućiti raspakivanje u originalni oblik samim programom PKLITE. [Preuzeto sa Sezama, autor **Junj Titov** (*jtitov*)]

497 [BASE.CLIPPER, PC] Mnoge početnike zbunjuju razlike između operatora = i ==. Razlika međutim nije samo estetska nego i te kako funkcionalna – znak = označava poređenje (*equal*) a == tzv. **puno** poređenje (*exactly equal*). Najbolje je da pogledate primere sa slike 4 – u zavisnosti od izabranog SET EXACT poređenja proizvode različite rezultate.

Operator == može da poredi i nizove. Ako ste, na primer, izvršile **a1:={1,2,3}**, **b1:={1,2,3}** i **a1:=c1**, **a1==b1** će dati vrednost FALSE a **a1==c1** vrednost TRUE. [Preuzeto sa Sezama, autor **Rade Vilimonović** (*vilb*)]

498 [OTXT WPERF 338, PC] U „Računarima 71" objavili smo prototip makroa koji omogućava da naša slova rade i sa pritisnutim *Caps Lock*-om. Međutim, kada sa tako definisanim makroima pređete u *Comment editor*, *Macro editor* i neke druge komponente *WordPerfect*-a, YU slova nema nigde. Problem je u tome što u tim editorima taster *Compose* (Ctrl V) ne radi, a rešenje je da se umesto sekvenci tipa {^V}1 99{ M} (slovo č) ubaci željeni kod komandom Ctrl 2 1 99 Enter. [Preuzeto sa Sezama, autor **Domen Ferbar** (*dferbar*)]

499 [MASM, PC] Na (praktično) svaki disk kontroler mogu se značno priključiti dva međusobno „kompatibilna" diska, ali može li se u isti računar priključiti više disk kontrolera, pa prema tome možda i desetak diskova? U većini slučajeva ne može, ali ima i kombinacija koje to obezbeđuju. SCSI kontroler, na primer, može da „koegzistira" sa sa AT BUS adapterom što je u praksi provereno za *Adaptec* SCSI kontrolere, a trebalo bi da važi i za druge. Izuzetak su neki jeftini *Seagate* SCSI adapteri koji nemaju mogućnost isključivanja flopi kontrolera. U računar se takođe može instalirati više *Adaptec* SCSI kontrolera, ali je to mogućnost koja se koristi uglavnom za *Novell NetWare*. U takvom slučaju treba konfigurisati drugi SCSI adapter tako da na njemu **nije** aktivan SCSI BIOS, sistem se docnije diže sa diska čija je adresa 0 na prvom kontroleru. Da bi *NetWare* video drugi disk, potrebno je instalirati mrežni softver uz upotrebu SCSI diskova. [Preuzeto sa Sezama, autor **Zoran Kehler** (*zkehler*)]

500 [IST, MASM] Videli smo kakve probleme diskovi danas donose, a sada da se podsetimo i starih vremena. Diskovi su nastali (iz zamenili magnetne cilindre i doboše) kada se neko dosetio da upisno-čitajuća glava može „lebdeti" nad površinom magnetnog medija na vazdušnom jastuku koji će nastajati običnim okretanjem ploča diska. Prvi diskovi su bili veliki kao veš mašine, a na dotične naprave su ličili i po izgledu kao i po zvukovima koje proizvode u toku rada. Glavni delovi su bili *spindle* (osovina-nosač koja drži magnetne ploče i koja je zaglođena je u lager), kočnica EMA (*Electro Magnetic Asembly* – kolica-nosač glava diska sa pratećim šinama), HDA (*Head Disk Assembly* – kućišta), elektronika za kontrolu, napajanje i nosači za sva ovo, u posebnom sanduku bio je disk kontroler a postojao je i poseban sanduk zvani *resistor terminator pack* koji se stavljao na kraj lanca diskova (danas ga „izigrava" nekoliko otpornika).

Diskovi su se delili na fiksne (FIX) i izmenjive (REM), operativni sistem je morao da se nalazi na fiksnom disku, a ako niste imali takav, mogli ste neki od izmenjivih diskova da proglasite za fiksni s tim da računaru i sebi date časnu reč da ga nikada nećete vaditi. Disk se delio na cilindre (sve glave po vertikali povezane), sektore (kružno po pločama / cilindrima, počevši od tzv. indeks markera) i klastere koji su zavisili od operativnog sistema. *Landing zone* je bio cilindar van korisne površine diska blizu njegovog centra, i bio je presvučen teflonom. Kontrolisano isključenje diska se izvodilo naredbama koje bi pozicionirale sve glave na *landing* sektore dok je za slučajeve prekida u napajanju postojao jedan poveći kondenzator zadužen da, napajajući potrebnu elektroniku, povuče sve glave u PARK poziciju. Pri „parkiranju" glave bi odlazile **van** landing zone i potpuno se izvlačile sa površine ploča, tako da više nisu u kontaktu sa diskom.

Disk se delio na particije, tako da je svaka aplikacija imala svoju particiju pa glave nisu morale da putuju (svaki sekund dobro dođe) sa *kraja na* kraj diska. Najveći neprijatelj diskova bile su vibracije, preterana vlažnost u sistemskoj sali i, naravno, aljkavi operateri. U slučaju mehaničkog potresa glava bi, pošto je bila **mnogo** veća od današnje, udarila po površini diska što je obično dovodilo do havarije poznate pod nazivom *head crach*. „Lečila" se zamenom par glava i čitavog *disk pack*-a. Zatvoreni (fiksni) diskovi nisu, kao što neki pogrešno misle, punjeni azotom, helijumom, vakuumirani i tome slično – unutra je bio čist vazduh pa je svaka jedinica morala da ima po nekoliko filtera, a zabranjivalo se i pušenje u blizini diskova. Dalji razvoj je najpre „uklonio" teflon u *landing* zoni, zatim su se glave smanjile, izmišljen je autopark koji „vozi" glave samo u *landing* zonu, 3.5-inčni diskovi su ukinuli i kočnice za *spinle*.

Do po zlu čuvenih *bad cluster*-a je uvek dolazilo iz istih razloga: greška u izradi magnetnog sloja na pločama, udarci glave po površini diska i tzv. zamor materijala – ako je sloj na nekom mestu našlo slabije izrađen, u početku je sve u redu ali se vremenom pojave problemi. Na nulti cilindar svakog diska (tamo su klasteri fizički najveći pa prema tome i najotporniji na oštećenja) upisuje se *boot sektor* i druge sistemske informacije, npr. FAT tabele pod DOS-om. Tu se takođe nalazi i *Bad Track Table* diska u koju su upisane pozicije loših klastera. Ukoliko se loši klasteri pojave docnije u toku rada, postoje programi koji će ih dodati u tu tabelu, neke zaštite se zasnivaju na tome da se savršeno ispravan sektor proglasi lošim. Ukoliko primetite rapidnu pojavu loših sektora iz dana u dan ili iz sata u sat, to **vrlo lako** može signalizirati neki kvar na elektronici diska koji će se ubrzo do kraja ispoljiti – požurite sa *backup*-om. [Preuzeto sa Sezama, autor **Zlatan Stojanović** (*zlaya*)]

501 [IGRE, PC] Ne baš najnovija ali svejedno veoma popularna igra *Pharaons tomb* ima mogućnost snimanja pozicije od koje se može nastaviti. Status se snima u datoteku PTOMB1.SAV. Prvi bajt (broji se od 1) tog fajla označava nivo, a deveti bajt je broj strela. *Checksum*-a nema, pa, na primer, izmena ovih bajtova na 3 i 55 izaziva početak igre od trećeg nivoa i sa 55 strela. [Preuzeto sa Sezama, autor **Dušan Popović** (*dusanp*)]

Priloge za ovu rubriku šaljite na adresu „Računari" (za „Bajtove lične prirode"), Bulevar vojvode Mišića 17, Beograd ili preko Sezama u vidu lične poruke uredniku (*mail write dejanr*)

INDEKSIRANJE BAZE

Nastavak sa strane 43

Bitno je navesti i broj sloga koji se briše jer stablo može sadržati više istih ključeva, pa funkcija ne bi znala koji da obriše.

Ukoliko prilikom brisanja ključa izbrišemo ceo čvor, mesto koje je on zauzima se ne oslobađa. Umesto toga se u datoteci čuva uvezana lista svih obrisanih slogova, tako da se sledeća alokacija mesta za čvorove ne vrši produživanjem datoteke već se nalaze prazna mesta iz liste.

- **void close (void),**

Zatvara indeks.

- **void keyval (void *ky),**

Vraća trenutni ključ.

- **RECNO currkey (void)**

Vraća trenutnu vrednost ključa.

- **RECNO nextkey (void),**
  RECNO prevkey (void),
  RECNO firstkey (void)
  RECNO lastkey (void),

Ove funkcije vrše sekvencijalno kretanje kroz indeks, kao i pozicioniranje na najmanji i najveći (tj. prvi i poslednji) ključ indeksa.

Sve ove funkcije vraćaju vrednost ERROR u slučaju greške. U protected delu klase se nalazi funkcija

- **virtual int compare_keys (void *a, void *b),**

koja služi za poređenje dva ključa. U osnovnoj implementaciji ova funkcija vrši poređenje stringova, međutim mi možemo želeti da koristimo neki drugi tip podataka za ključeve. Na primer, ukoliko bi smo imali stablo int-ova, napravili bi smo naslednika klase btree, recimo inttree, koji bi ovu funkciju realizovao kao

```
int inttree::compare_keys (void *a, void *b)
{
return *(int*)a - *(int*)b;
}
```

Još je samo potrebno sagraditi odgovarajuće stablo sa

build_tree (ime_stabla, sizeof (int)),

i sada možemo koristiti stablo sa **int**-ovima za ključeve. Takođe, u protected delu se nalazi i funkcija **dberror**, koja ima zadatak da prijavi grešku i onda ugasi aplikaciju. Greške koje mogu da nastanu su nedovoljno memorije i problem sa IO operacijama. Ova rutina se svodi na obično obaveštenje o grešci i naredbu exit, ali, kako je virtualna, vi je možete promeniti tako da u nekom, recimo, grafičkom okruženju otvori prozor i kulturno obavesti korisnika o grešci.

Funcija za rad sa diskom su takođe virtualne, tako da ih lako možete prilagoditi svojim potrebama. U klasi su takođe i dva konstruktora, jedan koji ne radi ništa i drugi koji poziva funkciju open. Destruktor poziva funkciju close ukoliko je to potrebno.

Ovo što imamo, dakle, predstavlja jedan univerzalan, ali sam po sebi potpuno funkcionalan kostur za pravljenje B-stabla. Stabla inače ne služe samo za baze podataka, primenjuju se kad god je potrebno brzo pristupiti nekom elementu po ključu. Setite se samo hipertaksta i helpova koje dobijate uz vaše prevodioce. Pritisak na Ctrl+F1 vas vodi na željenu stavku. Kako? Pomoću B-stabla.

Uzmite sa Sezama

Kompletan listing klase **Btree** može se preuzeti sa *Sezama* iz direktorijuma **rsoft** pod imenom **r083tree.zip**

Ni ova rubrika, očigledno, ne može izbeći uticaj pojava nove verzije OS/2 operativnog sistema. Verujem da su mnogi od vas, nakon instalacije OS/2, sa nestrpljenjem pokušali da generišu neki kod koji bi radio u pravom OS/2 režimu. Verujem, takođe, da je zamisao mnogih bila da provere da li se konačno pod načim što radi na Intelovom mikroprocesoru može deklarisati niz od, racimo, 2 MB sa istom onom jednostavnošću sa kojom se to radi kada niz ne prelazi 64 KB

OS/2

Kao što je napisano i u prikazu, za potpun i praktično proveren odgovor ćemo morati da sačekamo da do nas stigne IBM C kompajler verzija 2.0 koji je, bar koliko je meni poznato, u ovom trenutku jedini kompajler namenjen novom okruženju. Do tada, ostaje nam da probamo po malo sa onim što se može uraditi sa MSC kompajlerom 6.0. Ovo su osnovne oblasti u kojima programeri često traže najširu moguću kontrolu za koju nije dovoljan skup funkcija koje nudi standardna C biblioteka – brz ispis na ekran i očitavanje tastature.

Za one koji nisu do sade imali nikakvog dodira sa pisanjem programa za OS/2, prvo da kažem da pri pisanju i kasnijem generisanju koda nema ničega što nije poznato iz rada za DOS okruženje. Ukoliko imate program koji koristi isključivo standardan skup C ovskih funkcija, sve što treba da učinite je da pri kompajliranju dodate prekidač -Lp (Link Protected Mode). Možete dodati i prekidač -G2 kojim terate kompajler da koristi specifične instrukcije 80286 mikroprocesora, s obzirom da nema opasnosti da se takav program pokrene na nekom XT-u, a dobija se nešto na kompaktnosti i brzini izvršnog koda.

Šta se onda razlikuje? Razlikuje se praktično sve čim se pređe na oblast direktnog obraćanja operativnom sistemu, a posebno na oblast nestandardnih tehnika komunikacije sa periferijama (ekran, tastatura itd). Funkcije iz grupe _dos_xxxx (kao _dos_open, _dos_getvect itd) su potpuno ukinute, dok su one koje počinju prefiksom _bios_xxx zamenjene sa nekoliko novih skupova – jedan sa prefiksom **Vio** je posvećen radu sa ekranom, drugi nosi takođe jasan prefiks **Kbd**.

Prva linija svakog programa sada obavezno, nego ikada, treba da bude uključivanje **os2.h** zaglavlja – bez njega nema načina da se program do kraja prevede. Uključivanje ove datoteke je takođe donekle specifično – OS/2 skup funkcija je zaista ogroman pa bi stalno uključivanje kompletnog zaglavlja nepotrebno opterećivalo proces prevođenja. Umesto da se korisniku daju nazivi svih datoteka u kojima se nalaze definicije, koristi se drugačiji sistem – uvek se uključuje samo jedna datoteka (**os2.h**) ali se prethodnim deklarisanjem konstanti bira šta zapravo želimo da uključimo. Tako sekvenca

```
#define INCL_SUB
#include <os2.h>
```

obezbeđuje da kompajler dobije prototipe i definicije struktura potrebnih za rad s **Vioxxx** funkcijama. Ponekad se koristi i obrnuta logika – ako definišete konstantu INCL_NOPM, kompletan deo posvećen *Presentation Manager* će biti izostavljen. Pogledajte uostalom sami sadržaj **os2.h** – izuzetno je kratak i informativan za **#include** mehanizam koji se ovde koristi.

Funkcija kojom se direktnije čita tastatura i koja zamenjuje **_bios_keybrd** je

KbdCharIn(struct KBDKEYINFO * kbd, int WAIT, int HANDLE);

Parametrom WAIT (0 ili 1) možemo birati da li čekamo na pritisak ili tražimo da funkcija odmah vrati stanje tastature. HANDLE treba da bude uvek 0 (bar prema dosadašnjoj dokumentaciji, moguće je da u verziji OS/2 2.0 ovaj parametar dobija smisao), a funkcija vraća informaciju preko KBDKEYINFO strukture koja ima sledeći sadržaj:

```
typedef   struct KBDKEYINFO {
          UCHAR    chChar,
          UCHAR    chScan,
          UCHAR    fbStatus,
          UCHAR    bNlsShift,
          USHORT   fsState,
          ULONG    time
          } KBDKEYINFO;
```

U **chChar** ćete pronaći ASCII kod pritisnutog tastera, dok će njegov **scan** kod biti u **chScan**. **fbStatus** i **bNlsShift** su posvećeni statusu kada se koristi podrška nekom od nacionalnih jezika i tastature, dok su polja **fsState** i **time** svakako najinteresantnija. U prvom ćete pronaći stanje svih mogućih SHIFT tastera – zapravo u 16 bita objedinjenu informaciju koja je donekle kompatibilna sa onom koju smo do sada dobijali od BIOS-a – prvih osam bitova je identično po rasporedu, dok se u drugih osam nalazi informacija koja je ranije tretirana kao *extended* – odvojeno stanje levih i desnih ALT i CTRL tastera i status pritisnut/otpušten za LOCK tastere (CAPS, NUM, SRCOLL). **time** polje sadrži sistemsko vreme u milisekundama. Sve u svemu, nema posebno novih informacija, ali se zato sve one dobijaju jednim pozivom, što znatno olakšava precizno kontrolisanje tastature. Naravno, ne treba ni napominjati da ova funkcija daje stanje svih tastera na *enhanced* tastaturama (dakle i F11 i F12).

Čitanjem stringa se bavi funkcija **KbdStringIn**

KbdStringIn(char *buffer, struct STRINGINBUF *len, int WAIT, int HANDLE);

pri čemu je

```
struct   STRINGINBUF {
         unsigned lenght_of_buffer,
         unsigned lenght_of_returned_string,
         }
```

Postoji još dosta funkcija iz ove grupe, ali će ove dve, verujem, biti dovoljne da zadovolje vašu početnu znatiželju.

Diskusija o tehnikama za postizanje brzog ispisa pod OS/2 postaje potpuno bespredmetna, jer više ne postoji pojam video memorije – ona je pod potpunom kontrolom operativnog sistema i nedostupna programima. Nama jedino ostaje da koristimo **Vio** funkcije koje su, na sreću, realizovane sasvim solidno i daju zadovoljavajuće brz ispis. Osnovne funkcije iz ove grupe je

VioWrtCharStr(char *str, int len, int row, int col, int HANDLE);

Njeni parametri su karakteristični i za ostale funkcije iz ove grupe – string koji se ispisuje ne mora imati terminator nula jer se navodi njegova dužina, a u svim funkcijama se navodi i x,y pozicija. Na naziv funkcije se može dodati i **Att** (dakle **VioWrtCharStrAtt**) i pre poslednjeg parametra umetnuti pointer na *char* u kome se nalazi video atribut kojim se string ispisuje. Zanimljivo je postojanje funkcije **VioWrtNCell**

VioWrtNCell(char *cell, int repeat, int row, int col, int HANDLE);

cell treba da sadrži dva znaka – ascii kod i atribut koji će biti ponovljeni **repeat** puta od pozicije **row,col**. U skupu funkcija nalaze se i one za čitanje sadržaja ekrana, *scroll* u svim pravcima, ispis uz obradu dela kontrolnih kodova (**VioWrtTTy(char *bufer, int len, int HANDLE)**) i konačno funkcije za pozicioniranje i formiranje izgleda kursora. U video BIOS OS/2 ubačena je i ANSI emulacija – može se uključiti/isključiti funkcijom **VioSetAnsi(int state, int HANDLE)**.

Dakle, i ovde je situacija slična kao kod tastature – nema ničeg spektakularno novog, ali je sve sistematizovano i izvedeno sa povećanim komforom.

Da se ne bi ovaj za mnoge prvi susret sa elementima programiranja za OS/2 završio samo sa ovim funkcijama koje su ipak, mada u znatno siromašnijoj formi, poznate i od ranije, predlažem da prekucate i prevedete primer sa slike. Koristite **CL -Lp -G2 pr.c** i nakon toga pokrenite program u OS/2 FULL SCREEN režimu. Verujem da će rezultat izvršavanja biti interesantan i stvarno vas zainteresovati za prave novitete koje OS/2 donosi.

Listing 1

```
SLIKA

#define INCL_NOPM
#define INCL_DOS
#define INCL_SUB
#include <os2.h>
#include <malloc.h>
unsigned id1, id2;
long c1, c2;

int far T1(void)
{  char l[64];
   for( ; ; ) { sprintf(l, "T+ne COUNTER = %8ld", id1, c1++);
              VioWrtCharStr(l, strlen(l), 10, 10, 0); } }

int far T2(void)
{  char l[64];
   for( ; ; ) { sprintf(l, "T+ne COUNTER = %8ld", id2, c2++);
              VioWrtCharStr(l, strlen(l), 11, 10, 0); } }

main()
{  char far *stk1, far *stk2;
   c1 = c2 = 0;
   stk1 = (char far *)malloc(2048) + 2048;
   stk2 = (char far *)malloc(2048) + 2048;
   DosCreateThread(T1, &id1, stk1);
   DosCreateThread(T2, &id2, stk2);
   while ( !kbhit() ) ;
   exit(0);   }
```

MEMxxx

Dok (ako?) OS/2 ne postane naša svakodnevnica i time mi znatno olakša sastavljanje ove rubrike zbog mnogo noviteta koje donosi, da se ipak vratimo „onome od čega se živi". Običaj je da ovde predstavim bar jednu bibliotečku funkciju sa nekim detaljima koji nisu opštepoznati, pa da nastavim sa tom praksom.

Funkcijama iz grupe **strxxx**, kao što su **strcpy**, **strcmp** i slične, možemo završiti najveći deo posla kada su u pitanju stringovi. Sve ove funkcije se zasnivaju na činjenici da se stringom smatra svaki niz znakova koji se završava znakom 0. Pošto je nula „potrošena" na ovaj način, manipulacija nizovima karaktera koji treba da sadrže bilo kakav podatak među kojima je i vrednost nule bez posebnog značenja, mora se obaviti drugim setom funkcija. Zato u biblioteci postoji skup sa prefiksom **mem**

void memccpy(void *, void *, int, size_t);
void memchr(void *, int, size_t);
int memcmp(void *, void *, size_t);
void *memcpy(void *, void *, size_t);
int memicmp(void *, void *, size_t);
void *memset(void *, int, size_t);

Na osnovu poslednjeg dela naziva jasno je da funkcije predstavljaju ekvivalente odgovarajućih **str** funkcija, ali se uvek na kraju liste parametara nalazi i jedan **size_t** tipa (ovaj tip je najčešće jednak **unsigned int** tipu) – njime se ograničava dužina niza nad kojim primenjujemo funkciju. Ako izostavimo one očigledne primene **memxx** funkcija za manipulaciju binarnim podacima u baferu, one se najčešće koriste kao sredstvo za postizanje bržeg rada programa. Umesto da operaciju prebacivanja elemenata nekog niza „jedno mesto niže" izvodimo u petlji

```
for(i = 1; i<MAX_ELEM; ++i)
      data[i-1] = data[i];
```

kod najvećeg broja kompajlera se daleko brže izvodi

```
memcpy( data, data+1, sizeof(data[0])
*( MAX_ELEM-1 ) );
```

Ograda „kod najvećeg broja kompajlera" stoji zato što oni najbolji mogu da prepoznaju o čemu se radi u ovakvoj petlji i da sami optimizuju kod koji će se interno svesti na nešto vrlo slično **memcpy** funkciji. Dok kompajleri ne budu sposobni da nas na ovakav način uvek razumeju, ostaje da **mem** funkcijama postižemo znatna ubrzanja programa.

Jedna od funkcija iz ove grupe je posebno često u upotrebi i nezamenljiva je – funkcija **memset**. Ona će bilo koju memorijsku zonu popuniti zadatim znakom što je najkraći i najbrži način da se inicijalizuje sadržaj strukture ili nizova, na primer

```
char *lines[MAX_LINES];
struct FTG ft[MAX_FTG];

memset(lines,0,sizeof(lines));
memset(ft, 0, sizeof(ft) );
```

Ne treba izgubiti iz vida da, ukoliko na ovaj način inicijalizujemo sadržaj strukture (a ne niza struktura kao u primeru), treba obavezno navesti & operator

```
memset( &ft[0], 0, sizeof(ft[0]) );
```

Kao mali kuriozitet – ukoliko napravite grešku i izostavite ovaj operator, a od MSC tražite maksimalnu optimizaciju (Ox, u stvari bitno je da tražite generisanje **inline** funkcija) sačekaće vas interesantna poruka o grešci.

U ZNAKU BROJA 1992

74. pitalica će svakako ostati zapamćena kao jedna od težih – u predviđenom roku pristigio je 38 tačnih odgovora, a našlo se i pedesetak „onih drugih"...

A postavka je baš bila kratka – trebalo je pronaći **najveći** prirodan broj koji je proizvod nekoliko prirodnih brojeva čija je suma 1992. Za razliku od 74. Pitalice koja je takođe bavila brojem 1992 i koju je takođe predložio **Robert Paušić**, ovaj problem se „grubom silom" ne može rešiti – moramo da se oslonimo na matematiku pa ćemo zadatak rešavati u opštem slučaju tj. odrediti najveći prirodan broj **N** koji je proizvod nekoliko prirodnih brojeva čiji je zbir **K**. Zatim ćemo to rešenje konkretizovati za K=1992.

Pretpostavimo da je **K** veće od 1 (K=1 povlači N=1) i neka je

N=a[1]*a[2]*...*a[m]

traženi broj, pri čemu je

a[1]+a[2]+...+a[m]=K

Najpre ćemo dokazati da su svi brojevi **a[i]** manji ili jednaki 4. Pretpostavimo suprotno – neka za neko *i* važi **a[i]>4**. Tada je broj

a[1]*a[2]*...*a[i-1]*(a[i]-2)*2*a[i+1]*...*a[m]

veći od N, jer je **2*(a[i]-2) = 2*a[i]-4 > a[i]+a[i]-4 > a[i]**. No taj broj ima osobinu da mu je zbir faktora i dalje **K**, što je suprotno pretpostavci da je **N** najveći takav broj.

Dokažimo sada da je 2 najveći stepen dvojke kojim može biti deljiv sa **N**. U suprotnom bi **N** bio deljiv sa 2^n gde je *n* veće ili jednako 3. Obzirom da, prema dokazanoj teoremi, ni jedan od **a[i]** nije veći od 4, među **a[i]** su ili tri dvojke ili jedna četvorka i jedna dvojka ili dve četvorke. U svakom od ovih slučajeva postoji broj veći od **N** čiji je zbir faktora i dalje **K** – u slučaju **2+2+2** može se uzeti **3+3** jer $2^3 = 8 < 9 = 3^2$, u slučaju **2+4** se takođe može uzeti **3+3** jer **2*4 = 8 < 9 =** 3^2, a u slučaju **4+4** se može uzeti **3+3+2** jer 4^2 **= 16 < 18 = 3*3*2**.

Iz ove dve teorema proizilazi opis najvećeg broja koji je proizvod brojeva čiji je zbir **K**. Faktori su najviše dve dvojke ili jedna četvorka i ostale trojke ili jedinice (ako ih ima). Jedinice, naravno, ne dolaze u obzir jer se par **(1,a[i])** može uvek zameniti sa **(2,a[i]-1)**, što daje veći proizvod.

Dakle, ako je **K=3*L+2** (tj. daje ostatak 2 pri deljenju sa tri) faktori broja **N** su trojke i jedna dvojka, tj. **N=(3^L)*2**. Ako je **K=3*L+1**, faktori broja **N** su trojke i dve dvojke ili jedna četvorka, tj. **N=(3^{L-1})*4**. Konačno, u našem slučaju je **K=3*L** pa su svi faktori broja **N** trojke – **N=3^L** odnosno **N=3^{664}**.

Kurioziteta radi, na slici 1 dajemo kompletan ispis ovog broja dobijen primenom programa *Mathematica 386*. Nagrade su zaslužili **Petar Beličev** iz Beograda (24000 dinara), **Bratislav Veljković** iz Paraćina (18000 din) i **Đorđe Ljubičić** iz Beograda (12000 dinara). Čestitke za zanimljiva rešenja zaslužili su i *Biljana Srdanov, Milan Božić, Nenad Bunčić, Boško Koprivica, Jozef Kratica, Slavko Kuknka, Branko Obradović* i *Aleksandar Vasiljević*. Pomenimo međutim i zanimljiv prilog **Miloša Prvulovića** (nagrada od 18000 dinara za najbolje rešenje dostavljeno modemom) koji se jedini zapitao „a šta ako se u sumi ne dopuštaju jednaki elementi?". Nije se samo zapitao – razmotrio je čitav problem!

Može se naime pokazati da za svaki broj **X** veći od 4 važi da je proizvod **(X-t)*t** veći od samog broja **X** za svako **t** u intervalu **[2, X/2]**; na sam dokaz ovde nećemo trošiti prostor, ali ćemo reći da se zasniva na analizi kvadratne funkcije **X*t-t*t-X**.

Ovaj dokaz zapravo znači da svaki od brojeva treba, ako je ikako moguće, rastaviti po opisanoj shemi. Za maksimalno **N** treba iskoristiti sve male brojeve počev od 2, pa će niz činiti brojevi 2 3 4 5 6 7 ... Nevolja je, međutim, u tome što broj 1992 ne može da se napiše kao suma nekoliko članova ovog aritmetičkog niza – 1952. godina (zbir brojeva 2, 3, ... 62) je davno prošla, a 2015 (2+3+ ... +63) nije tako blizu. Dakle, traženih „nekoliko" brojeva je 61, a ostatak u sumi do 1992 može da se dobije uvećavanjem najvećih **a[i]** za po 1. Radi se dakle o brojevima 2...22 i 24...63 čiji je proizvod otprilike **8.62*10^{85}**.

Slika 1

```
2^996 = 66969287949141707559276565566250113160087800731595850465
        523439927314694069530850765582489867598099113297466705734
        707167657419658035576962772490360984186609252459104859265
        144365888171628163981963673721363845654046864738713292124
        229724478464966298164321606997798554088854787768644782890
        24177325354254336
```

PROBLEM GALAKTIČKE DECE

Nasuprot starom Veisu, Marsovci i Zemljani žive su u prijateljstvu radeći čak zajednički na rekonstrukciji marsovskih kanala. Nedavno je naš prijatelj Sven, inače jedan od inženjera kanala, doputovao sa Marsa gde ga je prijateljski ugostio glavni inženjer Nep, koji ga je čak upoznao sa svojim sinovima koji se zovu Lu, Ku i Bu. I Sven je ponosno pokazao sliku svoje ćerčice Stine, rekavši koliko ona ima godina. Nešto kasnije, naišao je i ministar saobraćaja doktor Baluk i, čuvši da se govori o godinama, upitao koliko su Lu, Ku i Bu stari, na šta je Nep odgovorio „Proizvod njihovih godina je 3520 a zbir je 14 puta veći od broja Stininih godina. Ne pravim, naravno, razliku između zemaljskih i marsovskih godina". Klimajući svojim glavama, doktor Baluk se požalio da i dalje ne zna koliko ko ima godina, na šta je Nep dodao „Vaš Labok je stariji od sve četvoro dece", na šta se ministar saobraćaja ponosno i zadovoljno osmehnuo.

Sven je mudro ćutao ali i dalje ne zna koliko Labok ima godina, pa po povratku na Mars može doći u neprijatnu situaciju koja bi ugrozila njegov planetarni ponos. Možete li mu pomoći?

Rešenja Pitalice, zajedno sa **obrazloženim** programima koje ste napisali, pošaljite na uobičajenu adresu „Računari" (za Dejanove pitalice), Bulevar vojvode Mišića 17, Beograd tako da pristignu **pre** 5. jula 1992. Sva pisma sa korektnim rešenjima konkurišu za novčane nagrade od 24.000, 18.000 i 12.000 dinara dok će kuponi (ili njihove fotokopije) na koje je upisan identifikacioni broj učestvovali u godišnjem takmičenju rešavača Pitalica. Identifikacioni broj dobijate tako što u prve tri kućice upišete **poslednje** tri cifre nečijeg broja telefona, u sledeće dve godinu vašeg rođenja i na kraj dodate dve cifre po izboru. Obratite pažnju da sva rešenja koja šaljete u toku godine nose isti identifikacioni broj. Osim običnom, rešenje ove Pitalice možete da pošaljete i elektronskom poštom posredstvom Sezama – koristite meniје REDAKCIJA, PITALICE, REŠENJE ili, iz komandnog moda, pošaljite ličnu poruku korisniku pitalice (*mail write pitalice*). Sva rešenja pristigla elektronskom poštom ravnopravno sudeluju u izvlačenju nagrada i konkurišu za **specijalnu nagradu** od 18.000 dinara koja će biti dodeljena najboljem rešenju koje je dostavljeno modemom!

76: Labok ima ________ godina

Ime i prezime ________

Adresa ________

Mesto ________

Kompjuter ________ Vreme ________

Identifikacioni broj |__|__|__|__ __ __ __

iz broja tel. god. rođ. po izboru

SEZAM

SISTEM ZA MODEMSKE KOMUNIKACIJE

Ne, ne vidite duplo: obzirom da prošlog meseca nije bilo prostora za ovu rubriku, ovoga puta objavljujemo dvostruki Bilten za mart i april! Moglo bi se reći da su oba meseca na Sezamu prošla u znaku stručnih razgovora — i pored uzbudljivih političkih događaja, diskusija kao da se preselila u konferencije čija imena počinju sa PC.

13

Priprema: Dejan Ristanović u saradnji sa moderatorima konferencija

Moderatori: Dejan Barnda (CIVILIZACIJA), Jovan Bulajić (PC USER), Aleksandar Damnjanović (PC OS), Slobodan Ka[illegible] (ORKA), Zoran Kehler (MOST), Vladimir Kostić (PC SOFT), Vladimir Krstinović (VICEV), Esad Jakupović (RAČUNARI), Darko Lanik (ATARI ST), Pavle Peković (PC PROG), Dejan Ristanović (NOVOSTI, KOMUNIKACIJE), Bojan Tepavčević (FORUM), Dejan Veselinović (PC HARD), Zoran Životić (SEZAM)

Kada je reč o bilansu, tokom marta i aprila „opslužili" smo 43 294 poziva, komunicirali sa korisnicima 9 952 časa, ili skoro 415 dana (obzirom da mart i april imaju 61 dan, u proseku je bilo **neprekidno** zauzeto 6,8 nodova) od čega je u *chat*-u provedeno 2680 časova ili 335 radnih dana. Korisnici su sa Sezama *download*-ovali 1,7 gigabajta podataka, dok je za isto vreme sistemu poslato oko 122 megabajta. U javne konferencije je pristiglo 14 536 poruka (preko 12 megabajta teksta), 7,6 megabajta datoteka priključenih uz njih, 13 760 privatnih poruka (preko 5 megabajta privatne pošte) i 6 206 poruka u grupama (2,6 megabajta teksta), ne računajući datoteke priključene uz te poruke.

U konferenciji **SEZAM** bilo je više poruka nego prethodnih meseci, i to ne samo zbog novih cena pretplate koje su prirodno izazvale dosta reakcija. Najaktivnija tema bila je *yu kod* u kojoj se diskutovalo o novom rasporedu naših slova (ćirilična kodna strana) na kome radi Savezni biro za standarde. Pošto su kolege iz Biroa izložile osnovne ideje, započela je detaljna i na momente vrlo oštra diskusija u okviru koje se, pored ostalog, i formirao konačni predlog standarda koji je radna grupa tokom aprila prihvatila; verujemo da će „Računari" o tome detaljnije pisati. U okviru ostalih tema trudili smo se da odgovorimo na uobičajena početnička pitanja o Sezamu i SOR-u i da prodiskutujemo sa korisnicima neka planirana unapređenja.

Konferencija **RAČUNARI** ima već ustaljen mesečni ritam – obzirom da su martovski „Računari" izašli na vreme, a aprilski nekoliko dana pre roka, komentara o zakašnjenju nije bilo ali se žalo diskutovalo o novom metodu lepljenja stranica (koji i dalje pokazuje izvesne slabosti), značajnom poskupljenju, naslovnoj strani, problemima sa dostavom časopisa van Beograda, pomalo čak i o sadržaju. Diskutovalo se i o konkurenciji na domaćem tržištu kompjuterskih časopisa, uz opšti zaključak da bi za „Računare" bilo bolje da imaju malo više konkurencije!

U konferenciji **KOMUNIKACIJE** diskutovalo se o *ViVa* modemima (koje Sezam koristi) i njihovim inicijalizacionim sekvencama (sa naročitim naglaskom na mogućnost prepoznavanja BUSY signala), priključenju dva (ili više) modema na PC računar, pulsnom i/ili tonskom biranju, *reed* relejima na modemima, ANSI sekvencama koje proizvode muziku, *fax* karticama koje omogućavaju automatsko uključenje čitavog računara u trenutku kada fax stiže, programu *Odyssey* koji je u prvi mah izgledao izuzetno dobro (naročito njegova softverska MNP emulacija) ali koji na kraju nije dobio naročito visoku ocenu, MNP *fossil* drajverima i njihovom „saradnjom" sa *Procomm*-om (uglavnom slabo sarađuju), *QModem* i *TransQ* protokolima, (domaćem) programu koji računa cenu telefonskih razgovora, sigurnosti koju postojeći 32-bitni CRC nudi (i pored njega desi se da neka „ispravno" prenesena datoteka bude oštećena), novim (i starim) domaćim BBS ovima koji nastaju i nestaju, stablu direktorijuma sa korisničkim programima na UBBG-u, oštećenim MAIL MAI datotekama na VAX-u, povremenim dugim putovanjima poruka preko *Bitnet*-a, novitetima na JUPAK-u i njihovim cenama.

Diskusija o *Stacker*-u u konferenciji **PC HARD** je polako privedena kraju (što ne znači da je više nema!) pa smo se vratili nekim starim temama kao što su cene računarske opreme, baterija na CMOS RAM-u, BIOS koji se štiti lozinkom, AMI *Setup* koji tek „iz druge" upisuje podatke, problemi sa QEMM-om 6.02 na nekim 80486 pločama, programiranje *Suntec* čipova, funkcionisanje hard diskova i njihovog „vazdušnog jastuka". Našli smo i neke zanimljive nove teme kao što su NCR 77C22E VGA kartice sa 2 megabajta RAM-a, priključivanje tri ili više disk jedinica na PC, nekompatibilnost aritmetičkog koprocesora i VGA kartica (znamo da zvuči sasvim, ali jednog korisnika je baš oko toga silno bolela glava), priključenje PS/2 monitora na ne-IBM-ove računare, *Cut Sheet Feeder* za *Epson* štampače, kvarovi laserskih štampača i tome slično. Možda ne bi bilo loše da i ovde prenesemo odgovor na pitanje koje je u ovoj konferenciji postavljeno bar desetak puta tokom zadnjih par meseci: hard disk NEC D3735 treba setovati na 2 glave, 1084 cilindra, 41 sektor.

U konferenciji **PC OS** mart je protekao u znaku *Unix*-a – bavili smo se raznim problemima vezanim za instalaciju štampača. Tema *ms dos* primila je uobičajeni broj poruka, a prenošene su i neke diskusije sa BIX-a o novom OS/2 2.0. U aprilu su te strane diskusije zamenjene domaćim – dobili smo prve (oduševljene) utiske korisnika o ovom programu uz napomenu da mu 4 megabajta RAM-a nisu dovoljna za komforan rad – potrebno je duplo više memorije. U okviru teme *unix* bilo je reči o konfigurisanju sistema za odloženu štampu, prepisu podataka na DOS diskete i situacijama kada sistem „misli" da ima dve video kartice, a dobili smo i prve vesti o verziji 4.0 operativnog sistema SCO UNIX. I tema *windows* je bila aktivna: razmenjivali su se prvi utisci o verziji 3.1, premda je bilo malo konfuzije oko prepoznavanja „prave" (komercijalne) varijante; saznali smo čak i kako da dodamo do (skrivenog) „grupnog portreta" autora (više o tome u prošlim „Bajtovima lične prirode"). Iako ova konferencija nije baš pravo mesto za takvu diskusiju, dosta poruka je bilo posvećeno programu *CorelDraw*.

U konferenciji **PC PROG** smo početkom marta zahvaljujući *edetic*-u konačno dobili toliko čekane informacije o MS BASIC-u 7.1, komentari se i dalje očekuju. Ostale teme su bile zatrpane porukama u kojima programeri traže pomoć za neke konkretne probleme. Tako smo naučili kako se povezuju asembler i paskal, kako „zaobići" probleme pri radu sa štampačem i nepravilnosti višestrukih relacija na *Clipper*-u. C programeri su se mučili sa *Borland*-ovim MAKE-om i sa formatom u *scanf* funkciji pa smo čak dobili i mali rečnik simbola koji se u okviru ove funkcije koriste. Sve je više interesovanja za objektno programiranje i to ne samo u C-u već i kod „klipeŕaša" koji su dobili biblioteku za rad sa objektima. Diskusija o zaštiti programa koja je počela u februaru ne samo da je nastavljena jednakom žestinom nego je dobila novu dimenziju jer se nije razgovaralo samo o tehničkom nego i o pravnom aspektu ovog problema. Da iskusniji programeri misle o svojim naslednicima, potvrdila je i rasprava o tome koje programske jezike treba izučavati u školama i koji od njih treba da bude prvi jezik. Aprilski dani su protekli u razgovorima o programiranju EGA/VGA karti, lovu na *bug*-ove funkcije *chrS* u raznim varijantama bejzika (čak i onim iz davnih vremena Spektruma, Commodore 64 i „galaksije"). „Klipeŕaši" su pokušali da nađu odgovor na pitanje o predrasudama u programerskom svetu o *Clipper* programerima i sastave listu želja koje bi trebalo da ispuni *Clipper 6.0* – traži se, pre svega, deklarisanje tipova promenljivih i kontrola tih tipova prilikom dodeljivanja i poređenja. I pored živih diskusija u pomenutim i nepomenutim temama, hit tema aprila je svakako bila *pascal*. Ljubitelji ovog jezika su diskutovali o prikazivanju PCX i GIF slika, uključivanju BGI-ova u izvršni program, *Turbo Vision* objektnoj biblioteci za TP, zatim pokušavali da kontrolišu miša, prikazuju istaknuti tekst na ekranu, kreiraju YU fontove... Posebno ističemo diskusiju o tipiziranim, netipiziranim i tekstualnim datotekama, tj. o poređenju mogućnosti i performansi rada sa njima, uz obavezno otklanjanje postojećih mana pomoću programerskih trikova.

Početkom marta je, sasvim očekivano, u konferenciji **PC USER** najaktivnija bila tema *virusi* – sve zbog „strašnog" Mikelanđela. Na sreću, korisnici Sezama su bili na vreme obavešteni i „naoružani" antivirus programima, tako da Mikelanđelo nije napravio mnogo štete – zabeleženo je nekoliko slučajeva zaraze (sa dosta teškim posledicama) kod prijatelja, ali se na Sezamu vrlo brzo našao pravi savet o tome kako povratiti podatke sa diska na kome se aktivirao Mikelanđelo. I tema *arhiveri* je bila dosta „živa" – pojava *HPack*-a je izazvala polemiku oko toga gde su „granice" u komprimovanju podataka, dok je alfa verzija PKZip-a bila povod (i) raspravi o imenovanju verzija komercijalnih programa – posle dosta priče zaključeno je da je to alfa, beta, gama... Programi za upravljanje memorijom su takođe bili vrlo interesantni za korisnike Sezama – najviše dileme bilo je oko korišćenja *stealth* opcije QEMM-a. Po temi *tekst procesori* moglo bi se zaključiti da i pored čuda zvanih *MS Word 4 Win* ili *WP 4 Win*, velikom broju korisnika nije potrebno ništa bolje od starog *ChiWriter*-a, ali se sve više njih raspituje o *TeX*-u. Početkom aprila ponovo smo se vratili temi *virusi*, ali znatno „dobroćudnijim" povodom – prvoaprilski SCAN koji je i na potpuno „čistim" računarima prijavljivao prisustvo opasnog virusa je mnoge prevario i uplašio, ali većina ipak nije nasela na provokaciju; u svakom slučaju, bilo je zabavno. Najčešće spominjan program bio je *WordPerfect*, a posebno njegova *Windows* verzija – otkrilo se da ovaj program mnogo pouzdanije radi sa *Windows 3.1*. Pričalo se i o problemima sa našim slovima, GRAB programu, drajverima za štampače, a i (prastari) WP 4.1 je preporučen kao najzgodniji za korišćenje na *laptop* računarima bez hard diska. Programi za upravljanje memorijom su bili drugi po aktuelnosti, posebno kada se radi o razlikama između *extended* i *expanded* memorije, i njihovom optimalnom korišćenju na AT računarima. U okviru ostalih tema bilo je priče o korišćenju softverskih emulatora koprocesora uz *Auto CAD*, programima za mrežno planiranje, čitanju bar koda, registrovanju nove verzije *Shez*-a, raznim uslužnim programima.

I u konferenciji **ATARI ST** bilo je dosta poruka

ČIPS BILTEN MART 1992

Konferencija	Poruka	Redova	Datoteka	Rekorda
UPUTSTVO	1	700	0	0
NOVOSTI	[illegible]	470000	[illegible]	920001
RAČUNARI	150	121885	0	0
ROBAN	[illegible]	601000	0	81500
KOMUNIKACIJE	710	[illegible]	11	[illegible]
PC HARD	[illegible]	280630	[illegible]	[illegible]
PC OS	100	100011	0	528240
PC PROG	[illegible]	531086	10	[illegible]
PC USER	071	340001	20	[illegible]
ATARI ST	100	183160	10	[illegible]
AMIGA	[illegible]	[illegible]	1	94810
ORKA	[illegible]	310000	10	[illegible]
FORUM	[illegible]	[illegible]	0	0
CIVILIZACIJA	100	140000	0	40000
EXTRA	[illegible]	[illegible]	0	[illegible]
CHAT	[illegible]	07100	0	0
VICEVI	416	082000	0	[illegible]
MALI OGLASI	[illegible]	[illegible]	0	0
HOST	55	40000	6	0
Ukupno	[illegible]	5785201	102	8258094
Privatne poruke:	[illegible]	[illegible]		
Grupe:	[illegible]	1394175		
Broj poziva				22481
On-line vreme				[illegible]
Chat vreme				[illegible]
File transfer vreme				[illegible]
Upload datoteka				1100
Upload kb				[illegible]
Download datoteka				19000
Download kb				[illegible]

Tema c se bavila *Turbo C*-om i njegovim PRJ fajlovima, ANSI standardom i dobrom literaturom. Tema *emulatori* je, kao i obično, posvećena *Mac*-u: pričalo se o prednostima HFS i MFS sistema, *PC-Acess*-u, softverskom PC emulatoru za *Mac*-a (koga je jedan atarista čak uspešno startovao na *Mac* emulatoru za ST) te o menjanju tipa i kreatora datoteka. Tema *grafika* je sada bogatija za opise dva nova programa za crtanje i animaciju – *Prism Paint* i *Chronos 3D*. Napokon je živnula i tema *dtp* kojoj je ozbiljno pretilo zatvaranje – poruke su uglavnom bile o *PostScript* izlazu ST/TT računara, prebacivanju *PostScript* fajlova na relaciji PC->ST->MAC i povezivanju ST-a na *Linotronic* foto slog. U najopštijoj temi *soft etc* se diskutovalo se o MiNT multitasking kernelu za ST-a (i šali na račun MiNT-a koji se reklamirao sloganom „MiNT is not TOS"; a pošto je novi MultiTOS zasnovan upravo na MiNT-u, slogan je promenjen u „MiNT is now TOS!"), MultiGEM-u i njegovoj budućnosti sa pojavom MultiTOS-a, GDOS fontovima i programu 7UP (ne, ne radi se o čuvenom bezalkoholnom napitku već o izvesnom tekst editoru). Dosta prašine je podigle vest o novoj verziji poznatog programa *Turbo ST*, ali se na kraju pokazalo da se neko samo igrao disk editorom. Ipak, sa najvećim oduševljenjem je dočekana nova verzija hard disk drajvera AHDI 5.0, koja više nema čuvenu opciju „check HD controller" koja je bez ikakvog upozorenja i mogućnosti prekidanja „testirala" kontroler tako što je brisala sve podatke na hard disku!

Da li se Vlada previše bavi računarima i da li su zakoni koji se tiču uvoza i izvoza KV režima u suštini štetni za razvoj računarstva? To je svakako bila tema meseca u konferenciji **ORKA** uz dosta predloga za peticiju, apel ili čak demonstracije korisnika računara. Nije se „ocenjivala" samo Vlada – igrači su poželeli da ocenjuju svoje omiljene igre, ako se u nekom od sledećih brojeva „Računara" pojavi Sezamova rang lista igara, značiće da je ideja urodila plodom. Tema *najbolji* se ponovo probudila, ali ovoga puta ne na relaciji PC-ST-Amiga nego između ljubitelja komandne linije i grafičkog interfejsa. Tema *hp48sx* donela je novi spisak PD softvera za ovaj računar, tražene su i nalažene knjige, komponente i saveti. Tokom marta konferencija se približila limitu od 5000 poruka, pa je početkom aprila otvorene nove, čiji je spisak tema ažuriran u skladu sa ispoljenim interesovanjima korisnika.

Mart je doneo nezapamćeno malo poruka u **FORUM**-u. Događaji kao što su 9. mart i demonstracije na Terazijama su prošli, u odnosu na prošlu godinu, skoro neopaženo. Diskutovalo se uglavnom o politici nekih stranih sila, njihovoj vojnoj moći i strateškim interesima. U okviru teme *novine* korisnici su razgovarali o raznim nezavisnim i „nezavisnim" glasilima i njihovom realnom uticaju, dok je tema *devojke* donela jednu prilično opširnu diskusiju o izvesnim „romantičnim doživljajima" na Sezamu. Diskusija se prenela i u april uz, za ovu temu sasvim uobičajeno, skretanje na mnoge stvari koje nemaju mnogo veze sa devojkama, no bilo je veoma napeto kada su se dve korisnice „dokačile" – bilans je preko 1000 poruka. U aprilu je bilo nešto više diskusije inspirisane političkim događajima u zemlji i oko nje, a sadržaj teme *Irak* bi svakako izazvao dosta interesovanja kada bismo ovde imali prostora da ga ponovimo.

U konferenciji **CIVILIZACIJA** bilo je, pomalo paradoksalno, najviše reči o nekim „statusnim pitanjima" tj. o temama *muzika* i *pokretne slike* i tome da li bi one trebalo da ostanu u konferenciji EXTRA ili da se „odcepe" i pređu u konferenciju CIVILIZACIJA. Inače, u samoj konferenciji je najživlje bilo u temama *ko je ko* i *o jeziku*. Pored ovoga, nepravedno bi bilo ne pomenuti interesantna literarna razmišljanja jedne nove, zanimljive korisnice koja je na prečac osvojila simpatije. April je u ovu konferenciju doneo „proleće" – gotovo da nema teme u kojoj nije bilo zanimljive rasprave.

KAKO POSTATI ČLAN

Postupak učlanjenja počinje zvanjem Sezama. Nakon poziva broja 648-888 i uspostavljanja veze (parametri 2400N8) i pozdravne maske, dobićete pitanje

Username

Na njega treba odgovoriti sa **NEW** (novi korisnik), nakon čega vas Sezam vodi kroz proces prijave. Prvi podatak koji treba da date je vaše puno ime i prezime, a potom birate pseudonim pod kojim ćete koristiti sistem. Pseudonim je bilo kakva reč do 16 znakova koja sadrži samo mala slova i znak . (tačka); obično se bira tako da asocira na vaše ime. Sistem će na osnovu vašeg pravog imena predložiti pseudonim koji možete ali i ne morate prihvatiti. Pre izbora pseudonima dobro razmislite, jer on postaje vaše trajno ime na Sezamu i ne možete ga više promeniti.

Sledeći korak je izbor lozinke (password) kojom se vaš pseudonim štiti – prijavljivanje sistemu je moguće samo ako poznajete lozinku koja je vezana uz vaš pseudonim.

Nakon ovih osnovnih podataka sledi izbor nekoliko radnih parametara (objašnjenja ćete dobiti na ekranu) a potom unos ostalih ličnih podataka – adrese, telefona i podataka o opremi koju koristite za zvanje Sezama. Ovim je prva faza gotova.

Posebnu obratite pažnju da **adresu unesete korektno** – ispravna adresa je jedini način da Uprava ostvari dalji kontakt sa vama kako bi se do kraja obavio proces vašeg učlanjenja.

Sledećeg radnog dana će vam biti poslato pismo u kome ćete pronaći i popunjenu uplatnicu. Na vama je da izaberete period na koji se učlanjujete i na osnovu toga na uplatnicu upišete iznos pretplate:

za 1 mesec	**3600 din**
za 3 meseca	**7000 din**
za 6 meseci	**12000 din**
za godinu dana	**20000 din**

Nakon izvršene uplate, fotokopiju potvrde o uplati pošaljite na naznačenu adresu Uprave Sezama bilo poštom ili (daleko brže) faksom. Po prijemu ove potvrde, Uprava će vam, istog radnog dana, odobriti pristup sistemu, čime postajete punopravan član Sezama. Istog dana će na vašu adresu biti poslato i Uputstvo za korišćenje sistema.

Ukoliko pri prvom prijavljivanju imate problema i ne završite popunjavanje podataka korektno, nije potrebno da zovete ponovo i prijavljujete se pod novim imenom. Sačekajte jedan dan pa pozovite ponovo – administratori sistema, kod detektovanja nepotpunih prijava, intervenišu i daju vam mogućnost da pri sledećem javljanju ponovo popunite prijavu ili vam ostavljaju ličnu poruku kojom vas obaveštavaju šta treba dalje da učinite.

Ukoliko ste se ipak prijavili pod više različitih pseudonima, imajte na umu da Uprava sama mora da izvrši izbor koju od vaših prijava će uzeti u dalju obradu. Kriterijum je najčešće kompletnost podataka. U slučaju više korektnih prijava uzima se u obzir samo poslednja. U svakom slučaju, na uplatnici će biti upisan pseudonim na koji se uplata odnosi i samo njega možete koristiti.

Ukoliko želite pre prijavljivanja da razgledate sistem, na pitanje Username odgovorite sa **DEMO**. Naći ćete se u demonstracionom režimu rada u kome možete pregledati kompletnu strukturu sistema koja je identična onoj pravoj. Samo DEMO pozivanje ograničeno, a vreme za jednu seansu je 5 minuta.

ČIPS BILTEN APRIL 92

Konferencija	Poruka	Redova	Datoteka	Rekorda
UPUTSTVO	[illegible]	[illegible]	[illegible]	[illegible]
NOVOSTI	[illegible]	[illegible]	[illegible]	[illegible]
RAČUNARI	[illegible]	[illegible]	[illegible]	[illegible]
ROBAN	[illegible]	[illegible]	[illegible]	[illegible]
KOMUNIKACIJE	[illegible]	[illegible]	[illegible]	[illegible]
PC HARD	[illegible]	[illegible]	[illegible]	[illegible]
PC OS	[illegible]	[illegible]	[illegible]	[illegible]
PC PROG	[illegible]	[illegible]	[illegible]	[illegible]
PC USER	[illegible]	[illegible]	[illegible]	[illegible]
ATARI ST	[illegible]	[illegible]	[illegible]	[illegible]
AMIGA	[illegible]	[illegible]	[illegible]	[illegible]
ORKA	[illegible]	[illegible]	[illegible]	[illegible]
FORUM	[illegible]	[illegible]	[illegible]	[illegible]
CIVILIZACIJA	[illegible]	[illegible]	[illegible]	[illegible]
EXTRA	[illegible]	[illegible]	[illegible]	[illegible]
CHAT	[illegible]	[illegible]	[illegible]	[illegible]
VICEVI	[illegible]	[illegible]	[illegible]	[illegible]
MALI OGLASI	[illegible]	[illegible]	[illegible]	[illegible]
HOST	[illegible]	[illegible]	[illegible]	[illegible]
Ukupno:	[illegible]	[illegible]	[illegible]	[illegible]
Privatne poruke	[illegible]	[illegible]		
Grupe:	[illegible]	[illegible]		
Broj poziva				[illegible]
On-line vreme				[illegible]
Chat vreme				[illegible]
File transfer vreme				[illegible]
Upload datoteka				[illegible]
Upload kb				[illegible]
Download datoteka				[illegible]
Download kb				[illegible]

Bilo je sjajnih istorijskih i arheoloških priloga, vođena je zanimljiva a često duhovita rasprava o jezičkim začkoljicama, dok je kraj meseca obeležila žestoka rasprava u temi *religija* – pokrenula su mnoga pitanja počevši od same religiozne potrebe pa do prakse. Pada u oči da je bilo i sasvim suprotstavljenih stavova, pa je pravo zadovoljstvo što je vođena izuzetno tolerantna diskusija u kojoj su se, bez ikakve arbitraže, poštovala i uvažavala tuđa mišljenja. Jednom rečju, ko nije prisustvovao ovoj konferenciji, propustio je priliku da nauči puno novih stvari u prijateljskoj atmosferi.

Što se konferencije **EXTRA** tiče, najviše prašine se podiglo u temi *muzika* kada se neko usudio da progovori o novokomponovanim narodnjacima. Našlo se tu žestokih kritičara i ponek simpatizer, a naročito je bila „opasna" korisnica koja je koristila pseudonim OKZŠS, što bi trebalo da znači „ona koja zna šta sluša" (zapravo, još je bolje znati šta **ne** sluša). U okviru teme *film* (za koju mnogi zahtevaju preimenovanje u *pokretne slike*) bilo je još odjeka na program FEST-a i predstavljanja raznih novih video kaseta. Diskusija u temi *biznis* se uglavnom bavila nekim udaljenim podnebljima, konkretno firmom *Microsoft* i njenim predsednikom Bilom Gejtsom (Bill Gates). Uobičajene „horoskopske" diskusije odvijale su se u okviru teme *zvezde*, dok je tema *mišljenje*, osim klasičnih, pokušavala da objasni nedavnu eksploziju nepoznatog tela nad Srbijom.

Konferencija **VICEVI** nikada nema razloga da se požali na manjak poruka. Pored najnovijih viceva koji redovno pristižu, pravo osveženje za konferenciju predstavljala je renesansa teme *aforizmi* kao posledica uključenja poznate beogradske aforističarke Vesne Denčić (*vecki*), koja nam je poslala dosta svojih aforizama kao i aforizama svojih kolega, što je podstaklo i druge korisnike da otvore svoje zbirke aforizama. Glasanje za vic meseca je gotovo rešavano „foto finišem" – pobedio je jedan prilično opširan vic o tome šta se kod žena menja sa godinama (zanimljivo je da je vic poslala jedna korisnica). Na primer: najveća laž je u 25. godini „brak nikad", u 35. „ljubavnik nikad", a u 45. „lifting nikad"; kad padne noć u 25. godini pitanje je „spavaš li?", u 35. „spavam!", a u 45. „spava (jadnik)"; upoznavanje počinje sa „zovem se", „da li se poznajemo", odnosno „sećaš li me se?"; uzajamno poveravanje je u 25. „recimo sve!", u 35. „nisi mi sve rekao", a u 45. „nisam te ništa pitala"... tabela je, rekosmo, prilično opširna! U glasanju za vic aprila trenutno vodi još jedan vic koji se bavi devojkama, a pomalo i računarima (zar nije izuzetna kombinacija?): „Šta dobijete kada plavušu ofarbate u crno? Veštačku inteligenciju."

kursora na dole izvršava se kodni blok **skipBlock** sa argumentom 1 (treba pomeriti kursor za jednu poziciju dole). Standardno ovaj blok izvršava **SKIP** naredbu i vraća vrednost „preskočenih" slogova. Što je najvažnije, programer može izmeniti ovaj blok (i još neke) i tako realizovati pregled proizvoljnog skupa podataka.

Na slici 4 je prikazan pregled jedne matrice stringova. Sada umesto funkcije **TBrowseDB** za kreiranje novog objekta pozivamo funkciju **TBrowseNew** koja takođe kreira objekt, ali bez unapred definisanih kodnih blokova za pomeranje kursora. Globalna promenljiva **I** označava tekuću poziciju vrste na kojoj je kursor. Ona u ovom slučaju zamenjuje pokazivač sloga kod DBF datoteke. Za pomeranje kursora treba zadati tri kodna bloka: za pomeranje na početak, na kraj i pomeranje za N redova.

Kodni blokovi za pomeranje na početak i kraj su najjednostavniji: u slučaju DBF datoteke oni jednostavno izvršavaju naredbe **GO TOP** i **GO BOTTOM**, a u ovom primeru je dovoljno da izmene vrednost promenljive I.

Kodni blok za pomeranje **skipBlock** ima nešto komplikovaniju strukturu: kada se on poziva, automatski mu se prosleđuje broj koji označava za koliko „slogova" treba pomeriti pokazivač. Situacija je jednostavna ako to pomeranje ne izlazi iz okvira podataka: ako je **I=5** i izvrši se pomeranje za dva sloga, na I se jednostavno dodaje 2 (matrica ima više od 7 redova). Ako se I ne može pomeriti za traženi broj, ovaj kodni blok je dužan da vrati ispravnu vrednost, tj. tačan broj slogova za koji je izvršeno pomeranje. Glavni deo ovog bloka realizovan je kao funkcija **MySkip()**.

Ovim nikako nisu iscrpljene sve mogućnosti objekata iz klase **TBrowse**. Korišćenjem osnovnih funkcija moguće je napraviti veoma složene tabelarne prikaze. U ranijim brojevima „Računara" bilo je govora o istovremenom radu sa dva **TBrowse** objekta: kreiraju se dva **TBrowse** objekta, i korisnik može paralelno kontrolisati oba.

Postupak se može uopštiti i za više od dva objekta: program na slici 5 omogućuje korisniku interaktivno kreiranje tabelarnih pregleda proizvoljnog skupa DBF datoteka. Tasterom <Enter> se definiše novi prozor (tabelarni pregled). Prvo se unosi naziv datoteke čiji pregled se traži, a zatim se definiše prozor u kome će biti ispisan prikaz. Prozor se pomera kursorskim tasterima, a veličina se menja tasterima <PgUp> i <PgDn>. Prelaz sa jednog na drugi prozor vrši se tasterom <Tab>.

Osnovna ideja je jednostavna: uvek kad korisnik zatraži definisanje novog pregleda, kreira se jedan **TBrowse** objekt i upisuje se u niz **aTB**. U niz **aFiles** paralelno se upisuje naziv oblasti u kojoj je otvorena odgovarajuća datoteka (jedna datoteka se može razgledati u više prozora). Promenljiva **nCurrent** pamti indeks trenutno aktivnog prozora, tako da program uvek radi sa objektom **aTB[nCurrent]** u radnoj oblasti **aFiles[nCurrent]**.

Kada se prelazi na drugi prozor, osim ažuriranja promenljive **nCurrent**, treba ponovo iscrtati okvir, što se jednostavno radi, jer svaki objekt sadrži informacije o koordinatama svog prozora (promenljive **nTop**, **nLeft**, **nBottom** i **nRight**). Pošto se jedna datoteka može razgledati u više prozora, neophodno je za svaki prikaz čuvati informaciju o broju tekućeg sloga u tom prikazu. Taj broj se može upisati u sam **TBrowse** objekt, u internu promenljivu **cargo**, namenjenu upravo takvim slučajevima: kada vam zatreba da neku informaciju vežete za objekt, upišite je u ovu promenljivu.

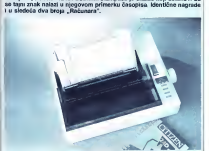
Mogućnosti **TBrowse** objekata zaista su neiscrpne. Korišćenjem ovih objekata možete praviti i neke sasvim neočigledne konstrukcije. Na primer, standardna funkcija **AChoice** se veoma jednostavno realizuje na ovaj način. Mogu se naći i razni tabelarni kalkulatori tipa *Lotus*-a (sasvim solidnog kvaliteta) realizovani kao dva preklopljena **TBrowse** objekta: u jednom se čuvaju formule za prikazivanje, a u drugom odgovarajuće vrednosti.

NOVA NAGRADNA IGRA "MERKOM" & „RAČUNARI"

Sačuvajte ovaj broj jer vam može doneti štompač!

SKRIVENI ZNAK ZA „CITIZEN"

Firma „Merkom" iz Kruševca u ovom broju „Računara" poklanja jedan štampač „Citizen" 124D. Dobitnik štampača je čitalac koji ustanovi da se tajni znak nalazi u njegovom primerku časopisa. Identične nagrade i u sledeća dva broja „Računara".



Štampač Citizen 124D

U ovom broju „Računara" na određenom mestu stavljen je u nasumično izabranom primerku tajni znak koji će vlasniku doneti na poklon 24-iglični štampač 124D poznate firme „Citizen". Znak ne treba tražiti odmah, jer je takve prirode da bi ga bilo veoma teško naći dok u sledećem broju ne objavimo o čemu se radi. Dakle, u sledećem broju objavljujemo obaveštenje o izgledu skrivenog znaka, kao i novi skriveni znak za drugog dobitnika nagrade. Vlasnik primerka sa skrivenim znakom treba odmah da se javi redakciji radi dogovora o proveri dobitka i predaji nagrade.

U slučaju da se primerak sa skrivenim znakom ne bude prodat nego se vrati BIGZ-u kao „remitenda", novi skriveni znak u narednom broju će biti otisnut u dva primerka časopisa – drugim rečima, nedodeljena nagrada se prenosi u novi broj.

U nagradnoj igri ne mogu da učestvuju zaposleni iz firmi BIGZ i „Merkom", niti njihovi bliski poslovni saradnici.

Firma darodavac tri štampača „Citizen" MERKOM iz Kruševca na domaćem tržištu štampača se „Citizenovim" programom pojavila se sredinom 1990. godine. Standardna ponuda se sastoji od 9-igličnih, 24-igličnih štampača svih klasa, preko termalnog notebook PN48 ink-jet-a, do vrhunskog profesionalnog PROLASER 12.

Godina 1990. i 1991. iskorišćene su za promociju ovog programa, koji je naišao na izvanredan odziv kupaca. Ono što firmu MERKOM razlikuje od sličnih je razvijena prodajno-servisna mreža, u koju su uključeni i takvi dileri kao što su „Šumadija", „Jugodata", „Jugolaboratorija", CET, „Tehnika", „Byte" i drugi.

Svetsko ime proizvođača, promptna isporuka, dvogodišnja garancija za sve proizvode, obilje dodatne opreme i tehnička ekipa kraj telefona 011/444-9705 i 037/28-386, uz izbalansiran odnos kvalitet/cena, garancija su pravilnog izbora.

(U sledećem broju detaljnije o „Citizenovom" programu štampača.)

DOVIĐENJA U SEPTEMBRU!

SLEDEĆI, SEPTEMBARSKI BROJ RAČUNARA

IZLAZI IZ ŠTAMPE KRAJEM AVGUSTA

OSTANITE I DALJE SA NAMA

Redakcija

Vladimir Stamenović

Oni koji rada sa nekom od Lotusovih tabela znaju da se, kada postoji puno datotaka na disku, gubi dosta vremana na tražanja imena ona koju treba učitati Program koji omogućava pregled raspoloživih datotaka, njihovo sortiranje po datumu, veličini, kao i po imenu bi mogao da predstavlja dragocenu pomoc Evo jadnog takvog programa koji, pored svega navedanog, nakon pragieda sadržaja diska, pruža i mogucnost učitavanja željene datoteke

Na slici 1 se vidi prostor gde će *1-2 3* ubaciti listing direktorijuma kao i nazivi makro procedura Prosirite kolone komandama **/Worksheet Column Set-Width** i to kolonu A na 13 kolonu B na 15 kolonu C na 12 i kolonu D na 11 zanakova Pomoću **/Range Format** komandi pridruzite ćelijama sledece formate **/Range Format Date 1 B4 B50 /Range Format, Date Time 3 C4 C50 /Range Format** (za raz ili Comma) **0 D4 D50**

Unesite tekst kao sa slike 1, s tim da unos u ćelije A1 i A2 počnete sa **Label Prefix**-om (obično apostrofom) Za unos duple linije ko jom je tekst podvučen ubacite bekslеš i znak jednakosti (\ =) u ćeliju A3 a zatim iskopirajte A3 u oblasl ćelija B3 D3 U ćeliju E4 unesite formulu @ISNUMBER(D4) pa je zatim iskopi rajte u oblast ćelija E5 E50 Ako direktorijum ima više od 47 datoteka pomenutu formulu iskopirajte nadole u koloni E, koliko je potrabno Kolonu E ćete „sakriti" sa displeja upotrebom komandi **/Worksheet Column Hide**

Da biste napravili makro procedure unesite podatke kao na slici 2 Unos podataka u ćelije J1 K2 J6 K7, J13 K13 i K14 počnite **Label Prefix**-om (obično apostrofom) Unesite i sle deće podatke

Ćelija	Tekst
K10	Sortiranje po datumu opadajući redosled
L10	Sortiranje po datumu rastući redosled
M10	Sortiranje po veličini opadajuci redosled
N10	Sortiranje po veličini rastući redosled
O10	Sortiranje po imenu datoteka

Upotrebite komande **/Range Name Create** da biste pridružili imena sledecim ćelijama ćeliju A4 nazovite DIME (od datoteka_ime) ćeliju B4 nazovite DDATUM a ćeliju D4 nazovite DBYTE Naziva makro procedura iz kolone J pridruzite odgovarajućim makroima iz kolone K pomoću komandi **/Range Name Label Right, J1..J15** Za kontrolu neka vam posluži slika 4 sa nazivima korišćenih oblasti ćelija i njihovim adresama Nakon toga snimite tabelu na disk

Kada pokranete *Lotus 1-2 3* učitajte datoteku sa makroima Startujte makro proceduru A (pritisnite i držite taster Alt i pritisnite taster A ili a pa oslobodite oba tastera) U prostoru koji je za to namenjen prema **slici 3** dobićete tabelu svih WK? datoteka iz TEKUĆEG direktorijuma kao i naziva eventualnih poddirektorijuma Ta oblast ćelija je definisana kao **Data-Range** a kolona E je definisana kao primarni ključ za sortiranje Da vidimo malo detaljnije kako makro procedura \ **A** funkcioniše Nakon aktiviranja, makro šalje pointer na početnu poziciju oblasti ćelija u kojima će biti prikazana tabela sa datotekama ({**HOME**}{**GOTO**}**A4**) Zatim poziva meni (/) pa redom komande **File Admin Table Worksheet** Na kraju ove linije imamo dve tilde (~ znak za enter) kojima potvrđujemo da će biti prikazane datoteke iz tekućeg direktorijuma kao i da je pointer na mestu gde će tabela sa datotekama biti postavljena

U drugoj liniji makro određuje **Data-Range** u kome će biti vršeno potrebno sortiranje pa ponovo poziva meni i komande **/Data Sort Reset Data-Range** U preostalom delu druge linije se odznacava oblast ćelija za **Data-Range** {.{**END**}{**D**}{**END**}{**R 2**}" što u „prevodu" znači ankersanje pointera taster END strelica na dole-DOWN taster END strelica u desno-RIGHT sve završavamo sa enter) U trećoj liniji makroa određujemo fiksni primarni ključ za sortiranje i otvaramo mogućnost da preko menija biramo sekundarni ključ Dakle komande za to su iz treće linije **Primary-key** E4 Enter D(escending-opadajuci), Enter Secondary-key Znači nismo odredili drugi ključ za sortiranje već takvu mogućnost predviđamo kroz meni odnosno preko makro procedure MMENI što se vidi iz četvrte linije makro procedura \ **A** ({**MENUBRANCH MMENI**})

Procedura MMENI će kreirati nas sopstveni meni sa komandama za sortiranje kako sam već na početku objasnio Dakle u kolonama K L M N i O u 9 redu su upisani nazivi komandi koje će biti u meniju dok su u 10 redu ispisane poruke-objašnjenja koja se pojavljuju ispod svake komande U 11 redu su makroi koji određuju sekundarni ključ za sortiranje jer upravo je tu makro \ **A** prepustio kontrolu daljeg toka proceduri MMENI Na primer (ćelija K11) na pitanje iz procedure \ **A** koji je **Secondary-kay** makro odgovara da je to ćelija DDATUM Enter D(escanding-opadajući) Enter I Go nakon čega se izvršava sortiranje

Ako poželite da ponovo sortirate tabelu sa datotekama pozovite makro \ S pa ćete dobiti meni iz koga obavljate posao

Na kraju ako hocete da učitate neku datoteku iz liste namestite pointer na ime te datoteke i pozovite makro ĐR Datoteka će biti učitana u *1-2-3* Da vidimo šta se tu događa Makro prvo iskopira naziv datoteke u ćeliju koju smo nazvali DATOTEKA Nakon toga izdaju se komande za poziv menija / pa **File Retrieve** Aktivira se taster Esc koji briše listu datoteka koju nudi 1-2-3 a makro se nastavlja imenom datoteke u ćeliji DATOTEKA (K15) i znakom za enter (K16) kojim prihvatamo to ime

Treba reći da će ovaj program raditi u *1-2-3* verzija *2 2* i višim Ako radite sa verzijom *3 x* morate napraviti malu izmenu u ćeliji K2

{ }**/DSRD** {**END**}{**D**}{**END**}{**R 2**

Ako poželite listing iz direktorijuma koji nije aktivan upotrebite **/File Directory** komande da ga aktivirate I još nešto da vas ne zbuni ako već imate jedan listing pa pravite drugi a on je kraći od prvog makro će ga upisati samo u onoliko redova koliko zahteva drugi listing

Ukoliko vam se ovakav način rada dopad ne možete makro nazvati \ **0** (nula) umesto \ **A** tako da se on automatski izvršava odmah po učitavanju tabele Tu još treba izdati i komande **Worksheet, Global Default Autoexec Yes Quit** Nakon toga snimite tabelu na disk Posle ponovnog učitavanja makro će se sam aktivirati

Ako imate nekih problema sa ovim primerom pokupite ga sa Sezama

Listing 1

```
        A           B

1 \e             Sortiranje direktorijuma
2 \r             Učitavanje datoteke (pointer na imenu datoteke)
3 ===========================================================
```

Nazivi makro procedura

Listing 2

```
        J         K          L          M          N          O

1 \e      {HOME}{GOTO}A4/FATW
2         /DSRD {END}{D}{END}{R 2}
3         PE4D~
4         {MENUBRANCH MMENI}
5
6 \s      {HOME}{GOTO}dime
7         /DRS{MENUBRANCH mmeni}
8
9 mmeni   Datum D    Datum R    Velicina D Velicina R Ime
10        Sortiranje Sortiranje Sortiranje Sortiranje Sortiranje po imenu datote
11        dDatumDG   DdatumAG   dbyteDG    dbyteAG    dimeAG
12
13 \r     /Cdatoteka
14        /FR{ESC}
15 Datoteka
16
```

Makro procedure

Listing 3

```
\e           Sortiranje direktorijuma
\r           Učitavanja datoteke (pointer na imenu datoteke)
=============================================================
AAJPTENT WK1      08 Apr 92     07 08 14       8 783
AMESTA02 WK1      08 Apr 92     08 42 01     118 006
DEPONIJE WK1      28-Jan 92     08 18 23       8 086
DIJACR23 WK1      20 Dec 91     12 14 28       7,101
DNE_PLAN WK1      08-Dec 91     10 05 00       6,748
JPIE WIN WK3      08 Apr 92     11 13 28       8 805
KARVEL01 WK1      12 Mar 92     11.54 09     143 075
KARVEL02 WK1      12 Mar 92     18 18 24     103,062
```

Datoteke sortirane po imanu

Listing 4

```
DATOTEKA  K15
DBYTE     C4
DDATUM    B4
DIME      A4
MMENI     K9
\0        K1
\A*       K1
\R        K13
\S        K6
```

Nazivi celija i oblasti sa adresama

SPECIJALNI KOMERCIJALNO-INFORMATIVNI BLOK
akcije o marketing o publicitet

OGLAŠAVANJE U „RAČUNARIMA" PRIBLIŽAVA VAS JAVNOSTI I OBEZBEĐUJE VAM NOVE KUPCE I KORISNIKE!

Primerak časopisa koji držite u rukama dokazuje da „Računari" imaju najkvalitetniju štampu, najbolji uvozni papir, najveći obim i – najkompletniju ponudu hardvera, softvera, periferija, prateće opreme, usluge konsaltinga, održavanja, obučavanja, poslovnih ideja, stručnog znanja i iskustva – u tradicionalnom komercijalno-informativnom bloku **„AKCIJE MARKETING PUBLICITET"**.

Kompletirajte tu ponudu svojom vlastitom – postanite i vi firma koju će mnogobrojni čitaoci „Računara" viđati na stranicama svoga časopisa

Cene oglasnog prostora se održavaju na zatečenom nivou u odnosu na stranu valutu, i nominalno se izražavaju u „bodovima" čija vrednost je nešto niža od vrednosti nemačke marke na slobodnom tržištu na dan fakturisanja

- 1/1 poslednja strana korica	**1 000 bodova**
- 1/1 druga strana korica	**600 bodova**
- 1/1 treća strana korica	**600 bodova**
- 1/1 kolor unutrašnja strana	**600 bodova**
- 1/2 kolor unutrašnja strana	**600 bodova**
- 1/1 crno-bela strana	**550 bodova**
- 1/2 crno-bela strana	**600 bodova**
- 1/4 crno-bela strana	**300 bodova**
- 1/8 crno-bela strana	**200 bodova**
- mali oglas (po centimetru)	**20 bodova**

Fakturisanje se vrši na dan ugovaranja oglasnog prostora i prijema potrebnih materijala za oglas sa obavezom da se uplata izvrši pre ulaska broja u štampu

Molimo vas da imate u vidu da je ROK ZA DOGOVOR I REZERVACIJU PROSTORA **25**, a za slanje oglasa **30** u mesecu, za broj koji izlazi sredinom narednog meseca

Ne zaboravite da možete obezbediti **odgovarajuće popuste** kod

– DUGOROČNIJEG OGLAŠAVANJA u „Računarima" (u 6, 9 ili 12 brojeva)

– PREDAJE POTPUNO PRIPREMLJENIH OGLASA (sa urađenim filmovima za kolor, odnosno filmom ili otiskom sa lasera za crno-beli oglas)

– PLAĆANJA UNAPRED za najmanje tri oglasa

Molimo vas da se za dogovore i informacije javite na
Telefon: 011/653-748, fax 011/648-140, modem 011/648-899
(osobe za kontakt: Vesna Jeremić, Esad Jakupović)

U OVOM BROJU VAM SE PREDSTAVLJAJU:

Srbiss	61	Proteus	69	Dibit	82
Aleksandrija	62	OTC	70	Comtec	83-87
Yuco	62	Elhard	71	KMA	88
Partner	62	Mraz	72/73	Most	89
Genel	62	BS procesor	72/73	ERC	90
Šutić	63	Preduzeće MZ	73	Hapel	91
CET	64	Comtrad	75	Imtel	92
Soft projekt	64	Ada	76	KMA	93
Pro muzika	65	KMA	77	CET	94
Info-D	65	Oracle	78	Compaq	95-97
Centar za per ol	66	Intersoft	79	Micro Air	98
Šutić	67	Digit	80	Mikronova	99
Penhard	68	OSA	81	Gama	100

INSTITUT ZA NUKLEARNE NAUKE "VINČA"
CENTAR ZA PERMANENTNO OBRAZOVANJE
11000 Beograd, Nemanjina 4/X
Telefon: 011/ 683-390, 682-486, 641-155/ 107,181
Telefax: 011/ 682-486

KNJIGE U OBLASTI PROGRAMIRANJA, PROGRAMSKIH JEZIKA I PRIMENE RAČUNARA

1. **AutoCAD (verzija 11.0)**
konstruisanje i projektovanje pomoću personalnih računara
Autori: Boris Damjanović i Petar Damjanović
Sedmo izdanje, izlazi iz štampe u junu 92.

2. **Uvod u C jezik**
Autor: Vladan Vujičić
Četvrto izdanje 1991 - latinica, 317 strana, format B-5, broširan povez

3. **Primena programa SYMPHONY na personalnim računarima**
Autor: Dragan Pantić
Treće izdanje 1990 - latinica, 226 strana, format B-5, broširan povez

4. **OS/2 - vodič za korisnike**
Autor: Zorica Jelić
Prvo izdanje, 1989 - latinica, 253 strane, format B-5, broširan povez

5. **VENTURA - računarsko izdavaštvo**
Autor: Predrag Davidović
Treće izdanje, izlazi iz štampe u junu 92.

6. **FORTRAN 77**
standard sa dopunama za personalne računare
Autori: Vlajko Kocić i Zoran Konstantinović
Drugo izdanje 1990 - latinica, 422 strane, format B-5, broširan povez

7. **UNIX - vodič za korisnike**
Autor: Zorica Jelić
Treće izdanje 1991 - latinica, 422 strane, format B-5, broširan povez

8. **Primena programa FRAMEWORK III na personalnim računarima**
Autor: Dragan Pantić
Drugo izdanje 1991 - latinica, 326 strana, format B-5, broširan povez

9. **Programski alati u matematici**
MathCAD, GRAPHER, EUREKA
Autor: Ante Čurin
Prvo izdanje 1990 - latinica, 402 strane, format B-5, broširan povez

10. **Primena programa QUATTRO na personalnim računarima**
Autor: Dragan Pantić
Prvo izdanje 1990 - latinica, 295 strana, format B-5, broširan povez

11. **DOS UKRATKO (verzije 3.3 i 5.0)**
Autor: Dragan Pantić
Drugo izdanje 1992. - latinica, 120 strana, format B-5, broširan povez

12. **Vodič za VAX/VMS**
Autori: Tamaš Kerepeš, Zvonko Oršolić, Saša Milijević
Prvo izdanje 1990 - latinica, 512 strana, format B-5, broširan povez

13. **Primena programa EXCEL na personalnim računarima**
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica, 272 strane, format B-5, broširan povez

14. **UNIX - vodič za programere**
Autor: Zorica Jelić
Prvo izdanje 1991 - latinica, 326 strana, format B-5, broširan povez

15. **WINDOWS 3.0**
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica, 273 strane, format B-5, broširan povez

16. **PRIMAVERA**
upravljanje projektima uz pomoć računara
Autori: Jaroslav Urošević i Jelica Drašković - Ostojić
Prvo izdanje 1991 - latinica, 365 strana, format B-5, broširan povez

17. **dBASE III + priručnik**
Autor: Milorad Filipović
Prvo izdanje 1991 - latinica, 249 strana, format B-5, broširan povez

18. **Osnovi informaciologije i informacione tehnologije**
Autor: Ljubomir Dulović
Prvo izdanje 1991 - latinica, 538 strana, format B-5, broširan povez

19. **LOTUS 1-2-3 (verzije 3.0 i 3.1)**
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica, 411 strana, format B-5, broširan povez

20. **dBASE IV priručnik**
Autor: Ljubomir Lazić
Prvo izdanje 1991 - latinica, 306 strana, format B-5, broširan povez

21. **WORDPERFECT (verzija 5.1)**
Autor: Dragan Pantić
Prvo izdanje 1991 - latinica, 300 strana, format B-5, broširan povez

22. **Programiranje u CLIPPER-u 5.01**
Autor: Alempije Veljović
Prvo izdanje 1992. - latinica, 465 strana, format B-5, broširan povez

23. **FoxPro**
Autor: Dušan Čalić
Prvo izdanje 1992 - latinica, 390 strana, format B-5, broširan povez

24. **Uvod u strukture podataka**
Autor: Miroslav Jocković
Prvo izdanje 1992. - latinica, 357 strana, format B-5, broširan povez

25. **ORACLE (verzija 5)**
arhitektura i administracija
Autor: Vladimir Milojković
Prvo izdanje 1992. - latinica, 151 strana, format B-5, broširan povez

26. **WORD za WINDOWS**
Autori: Nada Pantić i Dragan Pantić
Prvo izdanje, izlazi iz štampe u junu 92.

27. **MS WORKS - upravljanje poslovnim procesima**
Autori: Mirjana Nikolić i Miroslav Nikolić
Prvo izdanje, izlazi iz štampe u junu 92.

28. **PASCAL**
standard sa dopunama za personalne računare
Autori: Zoran Konstantinović i Slobodan Simić
Prvo izdanje, izlazi iz štampe u junu 92.

"RAČUNARI", jun 1992

Poručujem (ispod rednog broja knjige upisati broj poručenih primeraka)

Redni broj knjige	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.
Broj poručenih primeraka																												

IME I PREZIME
(Naziv preduzeća) ______________________
ULICA I BROJ ______________________
BROJ POŠTE I MESTO ______________________ TELEFON ______________

Porudžbenicu poslati sa petim primerkom uplatnice (informacije o cenama svakog radnog dana od 8 do 20 sati)
Uplata se vrši na žiro račun: INSTITUT ZA HEMIJSKU DINAMIKU I PERMANENTNO OBRAZOVANJE, broj 60803-609-698
Troškove isporuke knjiga snosi poručilac i namiruje ih unapred zajedno sa uplatom za knjige

PC AT 286, 386 SX, 386, 486 ISA - EISA

SC 2020	SC 2020 VGA	SC 3025 SX VGA	SC 3040 VGA CACHE 64 Kb	SC 4033 VGA CACHE 64 Kb	SC 4050 VGA CACHE 64 Kb
286-20 MHz **BABY 200W** **1Mb/MAX 4Mb** FD 1 2 Mb **HD 42 Mb 28 ms** **IDE - AT BUS** 2 SER +2 PAR TASTATURA 101 **HERCULES** **14" HERCULES** **REZ 720*350**	**286-20 MHz** **BABY 200W** **1Mb/MAX 4Mb** FD 1 2 Mb **HD 105 Mb 17 ms** **IDE - AT BUS** 2 SER +1 PAR TASTATURA 101 **SVGA CARD 512K** **14" SVGA MONO** **REZ 1024*768**	**386SX-25MHz** **BABY 200W** **1Mb/MAX 8Mb** FD 1 2 Mb **HD 105 Mb 17 ms** **IDE - AT BUS** 2 SER +1 PAR TASTATURA 101 **SVGA CARD 512K** **14" SVGA MONO** **REZ 1024*768**	**386-40 MHz** **TOWER 200W** **4Mb/MAX 32Mb** FD 1 2 Mb **HD 105 Mb 17 ms** **IDE - AT BUS** 2 SER +1 PAR TASTATURA 101 **SVGA CARD 512K** **14" SVGA MONO** **REZ 1024*768**	**486-33 MHz** **TOWER 200W** **4Mb/MAX 64Mb** FD 1 2 Mb **HD 105 Mb 17 ms** **IDE - AT BUS** 2 SER +1 PAR TASTATURA 101 **SVGA CARD 512K** **14" SVGA MONO** **REZ 1024*768**	**486-50 MHz** **TOWER 200W** **4Mb/MAX 64Mb** FD 1 2 Mb **HD 105 Mb 17 ms** **IDE - AT BUS** 2 SER +1 PAR TASTATURA 101 **SVGA CARD 512K** **14" SVGA MONO** **REZ 1024*768**
1.380 DEM	**1.050 DEM**	**1.980 DEM**	**2.650 DEM**	**3.450 DEM**	**3.950 DEM**

CENE DELOVA	1 - 10 KOM	11 - 30 KOM	PREKO 30 KOM
HARD DISK 42 Mb 28 ms	500 DEM	470 DEM	430 DEM
HARD DISK 105 Mb , NEC - 17 ms	**800 DEM**	**750 DEM**	**680 DEM**
HARD DISK 120 Mb , CONNER - 17 ms	850 DEM	800 DEM	720 DEM
HARD DISK 130 Mb , MAXTOR	**900 DEM**	**850 DEM**	**780 DEM**
HARD DISK 170 Mb , CONNER (serija JAGUAR)	1250 DEM	1150 DEM	1080 DEM
HARD DISK 200-210 Mb , 15 ms	**1500 DEM**	**1400 DEM**	**1280 DEM**
IDE - AT BUS 2HD/2FD CONTROLER + 2RS232	60 DEM	50 DEM	45 DEM
SCSI 2HD/2FD CONTROLER MAX 4 Mb CACHE	**550 DEM**	**490 DEM**	**450 DEM**
TOWER KUĆIŠTE + 200 W	250 DEM	225 DEM	199 DEM
AT BABY CASE + 200 W	**180 DEM**	**155 DEM**	**135 DEM**
TASTATURA MITSUMI 101	100 DEM	90 DEM	80 DEM
SVGA COLOR MONITOR 14 " 1024 x 768	**750 DEM**	**690 DEM**	**595 DEM**
HERCULES MONO MONITOR 14 "	250 DEM	220 DEM	190 DEM
SVGA MONO MONITOR 14 " 1024 x 768	**300 DEM**	**250 DEM**	**210 DEM**
SVGA COLOR MONITOR 19 "	2900 DEM	2700 DEM	2500 DEM
SVGA CARD TRIDENT 512 Kb/MAX 1Mb	**200 DEM**	**180 DEM**	**160 DEM**
1 Mb RAM (SIMM, SIPP)	110 DEM	100 DEM	85 DEM
ETHERNET NE 2000 16 Bit (NOVEL)	**380 DEM**	**360 DEM**	**330 DEM**
IIT 287-20 MHz MAT. KOPROCESOR	200 DEM	180 DEM	150 DEM
IIT 387/SX-25 MHz MAT. KOPROCESOR	**300 DEM**	**270 DEM**	**240 DEM**
IIT 387-40 MHz MAT. KOPROCESOR	500 DEM	450 DEM	400 DEM
FLOPPY DRIVE 1 44 Mb , 3 5 "	**150 DEM**	**140 DEM**	**130 DEM**
FLOPPY DRIVE 1 2 Mb , 5 25	165 DEM	150 DEM	140 DEM
BOARD 286-20 MHz	**200 DEM**	**180 DEM**	**160 DEM**
BOARD 386 SX 25 MHz	380 DEM	320 DEM	295 DEM
BOARD 386-40 MHz - 64 Kb CACHE	**750 DEM**	**690 DEM**	**650 DEM**
BOARD 486-33 MHz - 64 Kb CACHE	1600 DEM	1490 DEM	1390 DEM
BOARD 486-50 MHz - 64 Kb CACHE	**2200 DEM**	**1990 DEM**	**1850 DEM**
EPSON PRINTER LX 400 A4/9 PIN/YU SET	540 DEM	500 DEM	470 DEM
EPSON PRINTER LQ 570 A4/24 PIN	**780 DEM**	**750 DEM**	**690 DEM**
EPSON PRINTER LQ 1070 A3/24 PIN	1150 DEM	1060 DEM	990 DEM

DD 01.02.1991. PLACA SE PDREZ NA PRDMET 3% (ZA FIZIČKA I PRAVNA LICA)

UPLATA JE PO DNEVNOM KURSU DEM NA TRŽIŠTU U BEOGRADU
RADNIM DANOM od 9 do 17 h
ŽIRO RAČUN: 60815-601-64787

100 % AVANS
ISPORUKA ODMAH

GARANCIJA 12 MESECI - SERVIS I ODRŽAVANJE

TEL: 011/332-607
FAX: 011/345-126
BEOGRAD, 15.05.92.

PREDUZEĆE
ŠUTLIĆ & MIKRO DIZAJN

11000 BEOGRAD
Kasavska 32/I sprat

ELHARD

25238 KULA
V VLAHOVIĆA 46
Tel 025/722-740
Fax 025/723-433

KUTIJA ZA DISK (100 kom sa bravom) 5 25	25 5
KUTIJA ZA DISK DD40L 3 5"	17
BRACKET FDD W/CABLE&ADAPTOR	10 5
BRACKET FOR HDD 3 5	3 5
C7 CO PROCESSOR & CPU	
C80287 12 IIT	170
C80387 16 SX INTEL	382 5
C80387 25	595
C80387 33	595
CABLE PC TO PRINTER	14
CABLE POWER PC EURO TYPE	8 75
CABLE RS232 TO RS232 E M	17 5
CABLE RS232 TO RS232 M M	17.5
CARD AT MULTI I/O 1P/1S/1G	20 475
CARD AT MULTI I/O 1P/2S/1G	29 75
CARD DISK MULTI I/O STD XT	52 5
CARD DISK MULTI I/O SUPER 1 44	68 25
CARD EGA 8 BITS	113 75
CARD ETHERNET 8BITS	245
CARD ETHERNET 16 BITS	297 5
CARD HDD XT MFM	105
CARD HDD/FDD IDE 1 1	31 5
CARD HDD/FDD MFM 1.2	126
CARD MGP (HERCULES)	29 75
CARD VGA 16BIT 256K ONLY C&T	78 75
CARD VGA 1MB TSENG LAB	217
CARD VGA 512K EXP 1M TRIDEN	147
KUĆIŠTE MINI TOWER WITH 200W	245
KUĆIŠTE C2KL WITH 200W	201 25
KUĆIŠTE C2K WITH 200W	185 5
KUĆIŠTE TOWER L. WITH 220W	341 25
ČISTAČ GLAVE ZA FLOPI 3 5	4 55
ČISTAČ GLAVE ZA FLOPI 5 25	4 55
DATASWITCH BOX 3 WAYS	35
DATASWITCH BOX 4 WAYS	38 5

DISKETE

DISKETA BLANK 3 5" DSDD	19 25
DISKETA BLANK 3.5 DSHD	35
DISKETA BLANK 5 25 DSDD	10 5
DISKETA BLANK 5 25 DSHD	15 75
DISKETA MAXELL 1 2MB	25 375
DISKETA MAXELL 1 44MB	45 5
DISKETA MAXELL 360KB	17 5

FLOPI DISKOVI

FDD PANASONIC 3 5" 1 44MB	138 25
FDD PANASONIC 5 25" 1 2MB	140
FILTER MONITOR 14" NYLON	26 25
FILTER MONITOR 14" ANTISTATIC	35
FILTER MONITOR 14" STAKLENI	49

HARD DISKOVI

HD SEAGATE	
HD ST 351AX 42MB 245ms	472.5
HD3096A 3 5 80MB/ IDE	700
HD3144A 3 5 125MB/ IDE	875
ED 1230A 3 5 200MB/ IDE	1487 5
HD ESDI 320MB FH	3062 5
HD ESDI 660MB FH	3832 5
HD SCSI 320MB FH	3062 5
HD 760MB SCSI MICROPOLIS	4200
HD 1 53 GB SCSI MICROPOLIS	6650
HD MAXTOR	
HD 7040A 40MB	500 5
HD 7080A 80MB	787 5
HD 7120A 130MB IDE	945

RAM

I C 1MB 08x9 SIMM —08	102
I C 256KB-10x9 SIMM —08	35
I C 41000-08	12.25
I C 41256-08	4 375
I C 44256-07	14
K BOARD 101 KEYS MSW ENG MALA	66 5
K BOARD 101 KEYS JKSW ENG STD	84
K BOARD 101 KEYS JSW ENG STD NTC	89 25

OSNOVNE PLOČE

MBD 286-16G (10/22MHz)	175
MBD 286 20MHz	210
MBD 386SX 25 MHz	402 5
MBD 386DX 33C64B 55MHz	630
MBD 386DX 40 C64K	682 5
MBD 486-33 C64K 0RAM ETEO	1592 5
MBD 486DX 50 C64K 0RAM UMC	2362 5

MODEMI

E 1800 INT 240 Baud	175
E 1896 INT 2400/9600 Baud Send/Receive Fax	297 5

MIŠEVI

MOUSE FANCY F-860 (Drzac podloga pretvarac)	66 5
MOUSE FANCY A5 F 340 (Držač podloga pretvarač)	56
MOUSE WITTY C 800	38 5

MONITORI

MONITOR COMPAL MUL 0 28/1024×768 A	1085
MONITOR CTX VGA 0 28/1024×768A	866 25
MONITOR COMPAL VGA MONO SVGA	358 75
MONITOR YANJEN VGA 0 39 1024×768	714
MONITOR COMPAL 14 MONO	255 5
MONITOR 20 VGA 0 28/1024×768	3115

ŠTAMPAČI

ŠTAMPAČ HP LASER IIIP (1MB 4PPM 36FONTS)	3129
ŠTAMPAČ HP LASER III (1MB 8PPM 36FONTS)	4952 5
HP SCANJET 9195 W/CONTROLLER	3675
ŠTAMPAČ STAR NX 1000 II	542 5
ŠTAMPAČ OKI 183 A3/9PIN/120CPS	638 75
ŠTAMPAČ EPSON FX 1050 A3	1302
ŠTAMPAČ FUJITSU DL 3400	1356 5
POSTOLJE ZA ŠTAMPAČ (2 Kom noz)	14
TRAKA ZA ŠTAMPAČ STAR NX 1000 II	14
TRAKA ZA ŠTAMPAČ OKI MO 183	10.5
TRAKA ZA ŠTAMPAČ FUJITSU DL 3400	15 75
SCANNER WITTY MS 3000 400DPI OCR	243 25
TAPE BACK UP DJ 10 120MB INT	855 75
TAPE BACK UP DJ 20) 250MB INT	1137 5
TONER ZA HP II P	192 5
TONER ZA HP III	236 25
UPS 500 W	752 5
UPS 1KW	1417 5

KOMPLETI I

1 RAČUNAR AT 286 16MHz 0WS
— KUĆIŠTE AT BABY SA 200W ISPR
— MBD 286AT 16MHz (21MHz) Landmark
— RAM 1MB
— FLOPI PANASONIC 1 2MB 5 25"
— EDD/HDD KONTROLER IDE 1 1
— GRAFIČKA KARTICA HERCULES
— TASTATURA 101 KLICK MSW
— MONITOR COMPAL 1 4" B/W MONO

M P CENA 975 50 DM

2 RAČUNAR AT 286 16MHz 0WS
— KUĆIŠTE AT BABY SA 200W ISPR
— MBD 286AT 16MHz (21MHz) Landm
— RAM 1MB
— FLOPI PANASONIC 1 2MB 5 25
— FDD/HDD KONTROLER IDE 1 1
— GRAFIČKA KARTICA HERCULES
— TASTATURA 101 KLICK MSW
— MONITOR COMPAL 14 B/W MONO
— HARD DISK 40MB IDE 24ms

M P CENA 1 448 00 DM

3 RAČUNAR AT 386SK 25MHz
— KUĆIŠTE AT BABY SA 200W ISPR
— MBD 386SX 25MHz
— RAM 1MB
— FLOPI PANASONIC 1 2MB 5 25
— FDD/HDD KONTROLER IDE 1 1
— GRAFIČKA KARTICA HERCULES
— TASTATURA 101 KLICK MSW
— MONITOR COMPAL 14" B/W MONO
— HARD DISK 40MB IDE 24ms

M P CENA 1 675 00 DM



PERFEKCIJOM DO SAVRŠENSTVA
PC-AT 286/16 MHz
Kućište Baby AT
Osnovna ploča 286/16 MHz
Hard disk 45 MB
RAM memorija 1MB
Flopy disk 1.2 MB ili 1.44 MB
Monitor 14" TTL i Herkules karta
Tastatura 101 taster, YU/E
CENA: 1.490 DM
PC-AT 386/40 MHz 64 KB Cache
Kućište Mini Tower
Osnovna ploča 386/40 MHz
Cache 64 KB
Hard disk 45 MB
RAM memorija 4MB
Flopy disk 1.2 MB ili 1.44 MB
Monitor 14" TTL i Herkules karta
Tastatura 101 taster, YU/E
CENA: 2.490 DM
PC-AT 486/33 MHz 256 KB Cache
Kućište Mini Tower
Osnovna ploča 486/33 MHz
Cache 256 KB
Hard disk 45 MB
RAM memorija 4MB
Flopy disk 1.2 MB ili 1.44 MB
Monitor 14" TTL i Herkules karta
Tastatura 101 taster, YU/E
CENA: 3.390 DM
M/B AT 286, 16 MHz 200 DM
M/B AT 386, 33 MHz, Cache 64 KB 660 DM
M/B AT 386, 40 MHz, Cache 64 KB 720 DM
M/B AT 486 33 MHz Cache 256 KB 1 670 DM
HD 44 MB 22ms AT-IF Seagate, ST351A 520 DM
HD 44-68 MB 25 ms MFM-RLL Nec, D3142 650 DM
HD 85 MB, 19 ms AT-IF WD, AC280 740 DM
HD 125 MB, 15 ms AT-IF Seagate ST3144A 940 DM
HD 210 MB, 15 ms AT-IF Seagate ST1239A 1 670 DM
HD 385 MB, 14 ms SCSI Micropolis, 1684 07 2 900 DM
FD 1.2 MB, 5.25" 160 DM
FD 1.44 MB, 2.5" 130 DM
Monitor TTL 14" 220 DM
Monitor VGA mono 14", 1024x768 420 DM
Monitor VGA color 14", 1024x768 880 DM
TVGA 1024x768, 512 KB 130 DM
VGA Tseng 1024x768, 1 MB 220 DM
Modem Exterm 2400 Baud, MNP5 340 DM
Fax/Modem 9600/2400 Bps send/receive pocket 528 DM
Streamer Jumbo 60/120 MB 600 DM
Printer Epson LX400 9pin 150 cps, A4 520 DM
Printer Hyndai, 9pin 180 cps A3 590 DM
HP LaserJet 6st/min, 1MB 3 190 DM
HP LaserJet 8st/min, 1MB 4 790 DM
RAM karta za LaserJet IIIp/III, 1MB exp 4MB 260 DM
BS PROCESOR
BEOGRAD, Hadži Nikole Živkovića 2
telefon 011/180 739, 181-311 lok. 405
telefax. 011/ 180-739
NIŠ telefon 018/51-000, 331-654
NOVI BEČEJ telefon 023/772 153
ELEKTRONIK
MEmraz
MUNCHEN Schillerstrasse 22 / II
telefon (99) 49 89 59 59 20
telefax. (99) 49 89 55 35 87

PC AT 486-50 EISA

SC 4050 EISA

OSNOVNA PLOČA MICRONICS - Made in USA, CACHE 256 Kb, PROCESOR INTEL 486-50, RAM 8 Mb / Max 64 Mb NA PLOČI, EISA SCSI KONTROLER 4 Mb CACHE, HARD DISK 210 Mb MAXTOR, CONNER S3-SVGA CARD 1 Mb (*Šest puta brža od TSENG LABS ET-4000*), **14 " COLOR MONITOR 1024 x 768, FD 1.2 Mb + FD 1.44 Mb, TASTATURA 101, TOWER KUĆIŠTE**

9.950 DEM

HARD DISK 210 Mb, 15 ms ...... 1.500 DEM
HARD DISK 360 Mb, 15 ms ...... 2.400 DEM
HARD DISK 540 Mb, 15 ms ...... 3.900 DEM
S3-SVGA CARD 1 Mb ...... 550 DEM

	1 - 10 kom	11 - 30 kom	preko 30 kom
TOWER KUĆIŠTE + 200 W	250 DEM	225 DEM	199 DEM
AT RARY CASE + 200 W	180 DEM	155 DEM	135 DEM
HERCULES MONO MONITOR 14 "	250 DEM	220 DEM	190 DEM
SVGA MONO MONITOR 14 " 1024 X 768	300 DEM	250 DEM	210 DEM
SVGA COLOR MONITOR 14 " 1024 x 768	750 DEM	690 DEM	595 DEM
EPSON PRINTER LX 400 A4 / 9 PIN / YU SET	540 DEM	500 DEM	470 DEM
EPSON PRINTER LQ 570 A4 / 24 PIN	790 DEM	750 DEM	690 DEM
EPSON PRINTER LQ 1070 A3 / 24 PIN	1150 DEM	1080 DEM	990 DEM
EPSON PRINTER FX 1050 A3 / 9 PIN / YU SET	1350 DEM	1260 DEM	1170 DEM
EPSON PRINTER DFX 5000 A3 / 9 PIN / "heavy duty" / 535 z/s	POZOVITE		

POLO
Tel.: 081/13-202
Fax: 081/12 [illegible]
PODGORICA

DISTRIBUTERI

EMI
Tel./Fax: 081/34-425
PODGORICA

UPLATA JE PO DNEVNOM KURSU DEM NA TRŽIŠTU U BEOGRADU
RADNIM DANOM od 9 do 17 h
ŽIRO RAČUN: 60815-601-64787

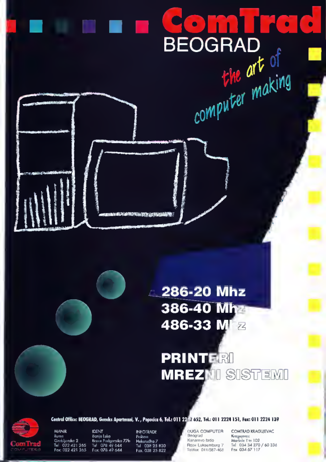
100 % AVANS
ISPORUKA ODMAH

GARANCIJA 12 MESECI - SERVIS I ODRŽAVANJE

TEL: 011/332-607
FAX: 011/34[illegible].126
BEOGRAD, 15.05.9[illegible]

PREDUZEĆE
ŠUTLIĆ & MIKRO DIZAJN

11000 BEOGRAD
Kosovska 32/I sprat


ComTrad
BEOGRAD
the art of
computer making
286-20 Mhz
386-40 Mhz
486-33 Mhz
PRINTERI
MREZNI SISTEMI
Central Office: BEOGRAD, Geneks Apartmani, V., Popocica 6, Tel.: 011 22?2 652, Tel.: 011 2224 151, Fax: 011 2224 139
ComTrad
COMPUTERS
MANIR
Ruma
Grobljanska 2
Tel 022 421 265
Fax: 022 421 265
IDENT
Banja Luka
Brace Podgornika 77b
Tel 078 49 644
Fax: 078 49 644
INFOTRADE
Priština
Hekurudha 7
Tel 038 25 830
Fax. 038 25 822
DUGA COMPUTER
Beograd
Kanarevo brdo
Roze Luksemburg 7
Tel/fax 011/587-468
COMTRAD KRAGUJEVAC
Kragujevac
Maršala Tita 102
Tel 034 34 270 / 60 336
Fax 034 67 117

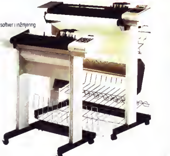
DPM - 160 Series ploteri

Vršimo usluge plotovanja i skeniranja.



Houston Instrument DPM-160 serija grafičkih plotera namenjeni za oblasti primene gde su produktivnost i preciznost najznačajniji

Kma: Vrhunske grafičke periferije, softver i inžinjering





Houston Instrument Summagraphics

kma d.o.o. 11070 N. Beograd • Milentija Popovica 9 • SAVA CENTAR

OVLAŠĆENI DISTRIBUTER ZA JUGOSLAVIJU I BUGARSKU

Tel 011/222-19-28
Tel 011/439-618
Tel 011/435-715
Tel 011/222-43-22 lokal 453
Fax 011/455-785 605-578

Ovlašćene prodajne organizacije.

SoftWell d.o.o.
Kompjuterski inžinjering
Pozorišni trg 1
21000 NOVI SAD
Tel 021/51 999 52 396

Lanaco
Braca Pavlica H
78000 BANJALUKA
Tel 078 39 088 39 162
Fax 078/33 992


DIBIT
DIBIT
DIBIT
MOUSE

MOST COMPUTERS

Beograd, Kralja Bodina 2, tel. 011 422 170, fax 011 426 144

BRUG COMPUTERS HOLLAND · BRÜCKE COMPUTERS GERMANY · MOST COMPUTERS YUGOSLAVIA

RAČUNARI • ŠTAMPAČI • SOFTVER • RAČUNARSKE MREŽE • DILERSKA PRODAJA

MOST JUNIOR	MOST EXPERT	MOST ASSISTANT
286/16MHz	*386/25MHz*	*386/33MHz*
1,490 DEM	***2,090 DEM***	***2,390 DEM***
MOST GENIUS	MOST GOLIATH[1]	MOST GOLIATH
386/40MHz	*486/50MHz*	*486/33MHz*
2,490 DEM	***4,390 DEM***	***3,590 DEM***

DOPLATE ZA VEĆE ZAHTEVE

Mono VGA monitor 'Supercom', SVGA grafička karta "OAK" OTI-037, 800x600/256kB	150 DEM
Kolor VGA monitor "Huyndai", SVGA grafička karta "Trident" TVGA 9000A, 1024x768/512kB	590 DEM
Kolor VGA monitor "Huyndai", SVGA grafička karta "Tseng" ET4000, 1024x768/1MB	745 DEM
Tvrdi disk 100-130MB	500 DEM
Tvrdi disk 200-210MB	900 DEM
Floppy disk 3 5"/1 44MB	159 DEM
RAM 1MB	125 DEM
Matematički koprocesor 287/12MHz	190 DEM
Matematički koprocesor 387/25MHz	380 DEM
Matematički koprocesor 387/33MHz	420 DEM
ŠTAMPAČI	
"EPSON" LX 850, 9-pinski, A4, YU set	690 DEM
"EPSON" LQ 570, 24-pinski, A4	790 DEM
"EPSON" LQ 1070, 24 pinski, A3	1,295 DEM
Laserski štampač "Hewlett Packard" HP LJ IIIp	3,450 DEM
Laserski štampač "Hewtett Packard" HP LJ III	4,800 DEM

Svi računari su apremljeni sa IDE tvrdi disk 'Microscience" 52MB sa internim cache-om, floppy 5 25" ili 3 5" "Supercom" monitor, "Hercules" grafički adapter, ASCII tastatura i uputstva na srpskom jeziku 286 i 386 računari: 1 ili 2MB RAM-a, mini tower kućište. 486 računari "Sayo" osnevna ploča, ETEQ chipset, 4MB RAM-a i tower kućište

Kupovinom računara »Most« dobijate sve što biste mogli da očekujete od računara firme sa velikim imenom, osim velike cene Pored toga, naši računari pokriveni su garancijom od godinu dana, i obezbeđenim servisom nakon tog roka.

Tehnička podrška – svakog radnog dana od 10-14h. HOTLINE 011-672-682.

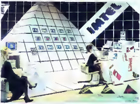
Ovlašćeni distributer – u Sava Centru, radnim danom od 9-17h. Tel. 011-222-43-22/492.



IMTEL je ime **IMTEL** je sinonim **IMTEL** je karika koja nedostaje



IMTEL-BABY-286	999
IMTEL-BUSINNESS-286	2220
IMTEL-286-16/40MB/Herkules	1449
IMTEL-286-16/40MB/VGA MONO	1499
IMTEL-486 po ceni 386	
IMTEL-LAN-286 MREŽA pet računara po 40MB	1299



štampači EPSON LX-800
BUBBLE JET BJ-10e
HP kolor printer
HP kolor skener
Multimedija (VIDEO+AUDIO+COMPUTER)
Windows akcelerator kartica
sva konfiguracije računara
komercijalni softver

IMTEL računarske mreže
ispletene tako da ništa ne promakne

IMTEL obuka
učite od nas, i mi smo učili od drugih
sad smo ih prevazišli

IMTEL poslovni softver (plate, računovodstvo...)
sva ostalo je guščije pero

IMTEL- Institut za Mikrotalasnu Tehniku i Elektroniku
Bulevar Lenjina 165 b, 11070 Novi Beograd
Tel. (011) 135-420,134-516 Fax: (011) 138-928,Telex:12797 IMTELYU.
Žiro račun 60815-603-14396

Priprema i uredjuje: „MICROSYS"

COMPAQ BILTEN

2
jun 1992

COMPAQ LTE Lite/20 & COMPAQ LTE Lite/25



Ekspanziona kutija za COMPAQ LTE Lite omogućava konverziju „notebook" računara u snažni „desktop" računar

Prema više sprovedenih istraživanja, korisnici „notebook" računara su rangirali svoje zahteve sledećim redosledom: mala težina, A4 format; visoke performanse, obezbedjenje od neovlašćenog korišćenja, uobičajena tastatura, racionalna potrošnja energije, veliki kapacitet magnetnih medija, brz modem, ekspanziona kutija, prihvatljiva cena

Veliki broj pomenutih problema je medjusobno čvrsto povezan Podjimo zato od jednog od njih, od racionalne potrošnje energije, koji je u suštini u osnovi svih navedenih zahteva Lepo je to što možete svoju omiljenu mašinu da vucarate okolo sa sobom, ali od nje nećete imati mnogo vajde ako svaki čas treba da joj pronalazite neku utičnicu da biste dopunili bateriju Pitanje je samo koliko se često ljudima desilo da im je, u avionu ili vozu, baš u trenutku dok su dovršavali svoju Harvard Graphics prezentaciju, njihov PC izdahnuo? Konačno, ovom Ahilovom petom se neko pozabavio Pre više od godinu dana Intel je najavio svoj 386SL čip, koji je obećavao mogućnost osmočasovnog trajanja baterija i 386SX/DX snagu Ali o tome se ništa nije čulo, sem neodredjenog mrmljanja vodećih „notebook" proizvodjača, sve do sredine prošle godine

U ovom tekstu vam predstavljamo dva člana COMPAQ-ove familije „notebook" računara, i to LTE Lite/20 i LTE Lite/25 Model Lite/20 je vrlo pogodan za tabelarna izračunavanja, obradu teksta i komunikacije (e_mail i slično) Model Lite/25 je pogodan za korisnike sa još ozbiljnijim zahtevima, za izvršavanje poslovnih aplikativnih paketa, finansijsku analizu, razvoj softvera, grafiku itd

Procesor 386SL

Šta, u stvari, nudi procesor 386SL? On ne radi baš na isti način na koji rade ostali Intel-ovi čipovi Zbog toga je 386SL mašinama i trebalo toliko da dospeju na tržište – one su tipski drukčije od ostalih iz porodice 80×86 Mnogo toga što se ranije nalazilo izvan procesora sada je integrisano u čip 386SL

U suštini radi se o sledećem čip 386SL nudi veću kontrolu nad hardverom sistema (hard diskom, procesorom, VGA grafikom itd) nego što je nude ostali Intel-ovi čipovi Najkraće rečeno, to znači da proizvodjači hardvera mogu iskoristiti ugradjenu kontrolu potrošnje energije, pružajući korisniku mogućnost izbora različitih opcija smanjenja potrošnje, uspavljivanja ili gašenja sistema

Drugi revolucionarni aspekt čipa 386SL leži u njegovoj visokoj integraciji Sastoji od dva glavna čipa, 80386SL, da mu navedemo puno ime, koji je u stvari procesorski čip sa memorijskim kontrolerom, kontrolerom magistrale i logikom potrebnom za podršku „cache" memorije i 82360SL, koji brine o I/O

Ali ovo je najmanje impresivno od svih 386SL dostignuća Najradikalnija novost koju donosi 386SL je mogućnost System Management Interrupt a (SMI) On kontroliše upravljanje svim periferalima, pa stoga i potrošnjom energije Ovo je naravno moguće i sa nekim drugim procesorima ali samo uz odgovarajuću softversku podršku Na taj način npr 386SX bazirani „notebook" računari ostvaruju kontrolu potrošnje Sa 386SL moguće je dizajnirati mašinu koja isključuje hard disk ali bez potrebe za TSR (Terminate and Stay Resident) programom Ovaj vam je stalno u mašini zauzima memoriju i vodi neprekidnu bitku sa Windows-om ili sa vašim omiljenim Memory Manager-om Intel je objavio da ima nameru da SMI ugradi u buduće procesore uključujući i 486 (486SL se prilično retko pominjao ali će se pojaviti ove godine ako sve bude po Intelovim planovima)

Konačno, 386SL je prvi Intel-ov čip kome se učestanost takta može spustiti na 0 Mhz Može se, dakle smanjivati brzina rada procesora (na polovinu četvrtinu itd) čime se u odgovarajućoj srazmeri smanjuje potrošnja energije Maksimalna učestanost rada ovog procesora je 20 odnosno 25 MHz

Neka zanimljiva svojstva LTE modela

Postoje četiri načina za kontrolu potrošnje energije Ovako efikasna i fleksibilna kontrola zahtevala je posebnu pažnju pri projektovanju podsistema baterije za napajanje koji, navodimo to kao zanimljiv tehnički detalj ima u sebi ugradjen 4-bitni mikroprocesor („smart" battery pack"), koji stalno prati potrošnju energije Racionalna potrošaja se postiže uz pomoć funkcije PWRCON (Power Conservation Utility), koja omogućava postavljanje željenih radnih uslova svih resursa računara, u skladu sa konkretnom aplikacijom korišćenjem jednostavnih grafičkih simbola (ikona) Primera radi ako se koristi Word/Windows sa srednjom potrošnjom može se podesiti da se ekran zatamni ako se nakon definisanog vremena ne pritisne ni jedna tipka, da se disk tiho „uspava" sve dok se ne zada komanda za smeštanje teksta na disk Na taj način se lako postiže 5 i više sati rada sa jednom baterijom

Hibernacija je tehnika kojom se kompletno stanje memorije (RAM) i stanje CPU registara smešta na disk i sistem se isključuje Ovo se može inicirati posebnom komandom a sistem automatski ulazi u ovo stanje kad ispražnjenost baterije predje odredjeni nivo Dakle, može se iz neke aplikacije, ne napuštajući je pritiskom na jednu tipku preći u stanje hibernacije i kasnije se vratiti na isto mesto

U „standby" režimu glavni podsistemi se isključuju a ostali se prebacuju u režim male potrošnje Ovo se može koristiti za kratkotrajne prekide do 48 sati jer se podaci čuvaju u RAM memoriji a ne na disku Korisno je npr da se, ne napuštajući trenutnu aplikaciju, izvrši zamena baterije, tako što se prethodno predje u „standby" režim

COMPAQ

Usporavanje sistema („system idle") predstavlja funkciju kojom se nakon definisanog perioda vremena u kome je sistem neaktivan, monitor zatamnjuje, a procesor usporava.

Na raspolaganju su 6 „vrućih tipki" koje omogućavaju podešavanje različitih funkcija za izmenu radnih režima i aplikacionog okruženja. Takodje je moguće pratiti potrošnju baterije (preostalo vreme se daje sa tačnošću od 10 min.), regulisati nivo zvučnika, izbor vrste zaštite i kontrolu video podsistema. Moguće je istovremeno korišćenje dva monitora, što je korisno u poslovnim prezentacijama.

Ekspanziona kutija je zanimljivi dodatak, koji omogućuje da na svom radnom mestu imate računar sa uobičajenim monitorom, dodatnim periferijama, mrezom (LAN) i slično, s tom razlikom što „srce" tog računara predstavlja vaš LTE model. Na kraju radnog vremena iskopčate LTE iz ekspanzione kutije i stavite ga u akta tašnu. Maksimalna sigurnost i mobilnost, pored ostalih već uobičajenih „MultiLock" tehnika koje COMPAQ standardno primenjuje (Power-on Password, Keyboard Password, Quicklock & Quickblank, Network Server Mode, Diskette Drive Control, Fixed Disk Drive Control, Fixed Disk Drive Password, Serial Interface Control, Parallel Interface Control, KeyLock, Cable Lock Provision (za ekspanzionu kutiju).

Kvalitet i kompatibilnost su svojstva koja se podrazumevaju kad su mašine COMPAQ u pitanju. Ovo ne samo da štiti investiciju kupca, već mu daje slobodu izbora od široko rasprostranjenih industrijski standradnih hardverskih i softverskih proizvoda. Korisnici imaju na raspolaganju servisnu podršku najvišeg nivoa. COMPAQ daje odgovarajuću garanciju na svoje proizvode, a uz pomoć mreže od preko 3 500 autorizovanih prodavaca, obezbedjuje odgovarajuću podršku u celom svetu.

Novi proizvodi – System Manager. Sa povećanjem složenosti lokalnih računarskih mreža, problem njihovog administriranja i održavanja se pojavljuje u sve ozbiljnijoj formi. Otkazi servera u tzv. „kritičnim aplikacijama" („mission-critical networks") mogu prouzrokovati velike štete. COMPAQ Computer Corp. je razvio poseban proizvod, tzv. SYSTEM MANAGER, koji predstavlja kombinaciju specijalizovanog hardverskog modula i odgovarajućeg softvera. Hardver (SM modul) se instalira na serveru mreže, dok softverski paket predstavlja kolekciju softverskih alata za daljinsko praćenje i nadzor mreže. SYSTEM MANAGER podržava *Novell NetWare V3.1X, Banyan Vines, Microsoft LAN Manager*, kao i *SCO Unix System V/386, Release 3.2X*.

Hardver System Manager-a sastoji se od 32-bitnog EISA modula, koji se instalira u server mreže. Na kartici se, pored ostalog, nalazi NiCd baterija, 2400 b/s modem, sintetizator glasa, serijski kanal za brzine prenosa do 9600 b/s, kao i naponski i temperaturni senzori za praćenje fizičkog okruženja računara. Baterija omogućava da SM modul nastavi da radi čak i kad server prestane sa radom, kao i da sačuva informaciju o serveru u trenutku otkaza. Kombinacija internog modema, sintetizatora glasa i



Računar COMPAQ LTE Lite/25

serijskog kanala obezbedjuje da se upozoravanje administratora mreže o stvarnim ili potencijalnim problemima može izvršiti na više različitih načina (na primer, slanjem poruke na pejdžer ili biranjem odgovarajućeg telefonskog broja).

Daljinski nadzor se vrši korišćenjem SMF softverskog paketa (*System Manager Facility*). SMF omogućava proveru stanja više servera na mreži sa jedne lokacije. Administrator može kontrolisati rad servera i izvršavati hladni start (*cold reboot*). Funkcija „udaljene konzole" omogućava uvid u monitor udaljenog servera i zadavanje komandi sa udaljenje tastature. Pored toga, može se pratiti statistika korišćenja memorije servera, stanje napajanja, temperatura i stanje podsistema diska.

Novi proizvodi – Familija računara SYSTEMPRO/LT. COMPAQ Computer Corp. je 3. marta ove godine predstavio novu familiju računara, namenjenu pre svega malim i srednjim grupama (5 – 40 korisnika) u LAN i UNIX okruženju. Reč je o EISA mašinama čije se performanse i resursi mogu podešavati prema konkretnim potrebama, a, sa druge strane, podsistem disk jedinica odlikuje se otpornošću na otkaze.

Modularni dizajn omogućava optimalno konfigurisanje sistema u skladu sa potrebama korisnika i to, kako u pogledu izbora procesora, tako i veličine memorije, vrste i kapaciteta diskova itd. Naknadni „upgrade" procesora je moguć zamenom procesorske ploče uz doplatu razlike u ceni izmedju prethodne (npr. 386) i nove (npr. 486) ploče. Korisniku su na raspolaganju 7 8/16/32 bitnih EISA priključnica za proširivanje osnovne konfiguracije.

U pogledu izbora vrste diskova, postoje modeli sa jednom jedinicom diska, kao i modeli sa inteligentnim poljem diskova (IDA – Intelligent Drive Array). IDA modeli, opremljeni robusnim 32-bitnim IDA kontrolerom i uparenim diskovima, obezbedjuju znatno bolje I/O performanse i otpornost na otkaze. Brzina prenosa podataka kod diskova sa IDA kontrolerom je oko 4 puta veća.

Postoje konfiguracije sa 5 različitih procesora: 386/25, 386/33, 486SX/25, 486/33 i 486DX2/50. U drugoj polovini 1992. godine se očekuje varijanta sa procesorom 486DX2/66. Kapacitet diskova se može birati u opsegu 120 MB do 1.02 GB, odnosno do 22.28 GB uz dodatak spoljašnje ekspanzione kutije.

Strateški planovi – Stop za RISC. Uzimajući u obzir promene korisničkih zahteva, Compaq Computer Corp. je odlučio da zamrzne svoje aktivnosti na planu razvoja komercijalnih sistema na bazi RISC (Reduced Instruction Set Computers) procesora u 1992. godini, kao i da se povuče iz ACE (Advanced Computing Environment) Inicijative, gde je bio je jedan od suosnivača, početkom 1991. godine. U toku poslednje dve godine Compaq je usmerio značajne razvojne i marketinške snage u cilju realizacije komercijalnih sistema na bazi RISC procesora. Ovo je bilo motivisano nedostatkom robusnih sistemskih okruženja za Intel-ove arhitekture i ukupnom razlikom u pogledu odnosa cena/performanse izmedju sistema na bazi RISC procesora i sistema na bazi predvidjanih novih Intel-ovih procesora. U medjuvremenu je došlo do



Standardne konfiguracije

COMPAQ LTE Lite/20	procesor 386SL/20MHz 1 44MB flopi postoje verzije sa diskom od 40 MB i 84 MB
COMPAQ LTE Lite/25	procesor 386SL/25MHz 1 44MB flopi postoje verzije sa diskom od 84 MB i 120 MB

Opcije

Ekspanziona kutija	omogucava jednostavnu konverziju „notebook računara u „desktop" model
Masovne memorije	5 25" flopi (360kB i 1 2MB) magnetna traka (60 i 80/120 MB) sa ekspanzionom kutijom može bilo koja dva od sledećeg 5 25" flopi diskovi 84MB 120MB 210MB traka 60MB 80/120MB CD ROM
Komunikacije	interni modem 2400b/s poboljšani interni modem 9 600 b/s
Grafika	14" mono ili kolor VGA monitor
Napajanje	mrežni adapter (220V/50Hz) automobilski adapter

Specifikacije

Lite/20	podnožje za koprocesor 387SX
Lite/25	procesor je sa keš kontrolerom i 16 kB 4-way set associative" keš memorijom podnožje za koprocesor 387SX
Kompatibilnost	potpuna sa industrijski standardnim personalnim računarima
Magistrala	ISA
Memorija	2 MB (max 10 MB) za Lite/20 odn 4 MB (max 12 MB) za Lite/25.
Napajanje	poboljšana NiCd za Lite/20 odn poboljšana NiMH (Nikl metal hidrid) za Lite/25.
Proširenja diskova	uz dodatne spoljašnje diskove max 294MB (Lite/20). odn max 330MB (Lite/25)
Interfejsi	serijski paralelni spoljašnja tastatura miš. spoljašnji VGA monitor
Grafika	VGA LCD monitor 16 nivoa sivog rezolucija 640×480 ekran 9 5"
Softver	COMPAQ-ovi uslužni programi CEMM ADAPT Disk Cache EZ help PWRCON
Tastatura	Kompatibilna sa modelom od 101 dirke
Zaštita	10 standardnih COMPAQ-ovih mera zaštite od neovlašćenog korišćenja
Mere	28 x 22 x 4 5 cm masa 2 7 kg

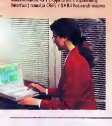
značajnih promena Intel-ova dominacija na tržištu procesora i agresivnost u razvoju novih procesora kombinovano sa prenosom (portiranjem) operativnih sistema novih generacija na Intel-ove platforme kao i kašnjenje u razvoju nove generacije RISC procesora uticala je da potencijalni korisnici značajno smanje svoje interesovanje za RISC sisteme Poštujući zahteve svojih dosadašnjih i budućih korisnika Compaq će se u 1992 koncentrisati na realizaciju širokog spektra vrlo konkurentnih otvorenih sistema na bazi Intel-ovih procesora koji imaju odličan faktor cena/performanse zajedno sa poboljšanom korisničkom podrškom (distributivna mreža održavanje i sistemska podrška)

Compaq je isto tako odlučio da se povuče iz ACE inicijative u skladu sa odlukom da odustane od RISC baziranih proizvoda kao i zbog namere članova ACE inicijative da se usmere isključivo na korišcenje MIPS hardvera Ipak ACE inicijativa je bila glavni katalizator promena u industriji Primera radi najava dogovora iz prošlog novembra o konzistentnom API (Application Programming Interface) između OSF1 i SVR4 baziranih sistema



System Manager

pozitivno je uticala na celokupnu UNIX industriju Sa druge strane svi budući Compaq-ovi proizvodi na bazi Intel-ovih procesora će zadovoljavati zahteve ACE inicijative definisane aprila 1991 Dakle biće kompatibilni sa novim operativnim sistemima Microsoft-a SCO-a i drugim novim robustnim operativnim sistemima koji tek treba da se pojave (Solaris NeXTStep Univel)



Strateški planovi – zaštita životne sredine Compaq Computer Corp je objavio da započinje saradnju sa američkom Agencijom za zaštitu prirodne sredine u cilju definisanja tehničkih zahteva u sklopu „Energy Star Computers" programa kojim se proizvodjačima računara preporučuje da projektuju i proizvode opremu koja će imati racionalnu potrošnju električne energije Pored toga Compaq je već ranije objavio svoj program za efikasnu i ekološki bezbednu reciklažu baterija iz svojih „laptop" i „notebook" računara

Ocene drugih Računar COMPAQ LTE Lite/25 je dobio PC USER GOLD AWARD priznanje koje se dodeljuje za proizvode koji se ocenjuju prema tehničkim karakteristikama kvalitetu dokumentacije podršci kao i ukupnoj vrednosti koja se dobija za uloženi novac (Objavljeno u časopisu „PC User" 26 Feb -10 March 1992)

Čitaoci časopisa „System Integration Magazine" su po ocenjivanju notebook" računara najviše glasova dali sistemu COMPAQ LTE 386s/20 Pored toga COMPAQ SYSTEMPRO 486 je proglašen za najbolji multiuser i fajl server računar a COMPAQ DESKPRO 486/33L i COMPAQ DESKPRO 386s/20 su dobili najviše ocene u svojoj klasi Na kraju COMPAQ Intelligent Array Expansion System je bio finalista u kategoriji vinčester i fiksnih diskova (Objavljeno u broju za decembar 1991)

Urednici časopisa „PC/Computing Magazine" su izabrali COMPAQ DESKPRO/M familiju modularnih personalnih računara za „The Most Valuable Product of 1991 "

Računar COMPAQ LTE Lite/25 je dobio priznanje „Editor s Choice" za najbolju kombinaciju performansi prenosivosti i cene koje daje časopis „PC Magazine" U poredjenju sa 14 drugih testiranih sistema ovaj model je prevazišao svoje konkurente u brzini zaštiti od neovlašćenog korišćenja i racionalnoj potrošnji energije (Objavljeno u časopisu „PC Magazine" od 14 04 1992)

Kontakt adresa

MICROSYS
21000 Novi Sad
Tel 021/611-366 Fax 021/20-544

Kontakt adresa

MICROSYS – Poslovnica Beograd
11000 Beograd Dobračina 39
Tel 011/625-994 Fax 011/622-581

GAMA Electronics



No. 1